NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

A MEASUREMENT STUDY OF BGP BLACKHOLE ROUTING PERFORMANCE

by

Nikolaos Stamatelatos

September 2006

Thesis Advisor:	Geoffrey Xie
Second Reader:	J. D. Fulp

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 2006	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE A Measurement Study of BGP Blackhole Routing Performance	**5. FUNDING NUMBERS**
6. AUTHOR(S) Nikolaos Stamatelatos	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

BGP Blackhole routing is a mechanism used to protect networks from DDoS attacks. During the last several years, a number of variations of BGP Blackhole routing have been proposed. However, even though these methods have been used by many organizations and ISPs for some years, the academic community has provided only a limited evaluation of BGP Blackhole routing, using mainly network simulations.

The objective of this research was to evaluate the basic methods of BGP Blackhole routing in a real test-bed network in various environments. By using the response time, the CPU load, and the link load as performance metrics, we first evaluated the performance of those methods in networks where the routers' CPU load was the limiting factor. Then we examined the effect of the high link load and the effect of routers' preconfiguration on the BGP Blackhole routing's performance.

The results showed that the BGP Blackhole routing may not be effective under stressful situations, that is, a high link load, because its dependence on TCP and the underlying routing protocols. Of the three basic Blackhole routing methods, the best method is the destination-based, followed closely by the source-based. The third method, customer-triggered Blackhole routing, in all cases had very degraded performance.

14. SUBJECT TERMS BGP, Blackhole Routing, Null Routing, DDoS Attacks, Network Security			**15. NUMBER OF PAGES** 111
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

THIS PAGE INTENTIONALLY LEFT BLANK

A MEASUREMENT STUDY OF BGP BLACKHOLE ROUTING PERFORMANCE

Nikolaos Stamatelatos
Captain, Hellenic Air Force
B.S., Hellenic Air Force Academy, 1993

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS ENGINEERING
and
MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
September 2006

Author: Nikolaos Stamatelatos

Approved by: Geoffrey Xie, Ph.D.
Thesis Advisor

J.D. Fulp
Second Reader

Dan C. Boger, Ph.D.
Chairman, Department of Information Sciences

Peter J. Denning, Ph.D.
Chairman, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

BGP Blackhole routing is a mechanism used to protect networks from DDoS attacks. During the last several years, a number of variations of BGP Blackhole routing have been proposed. However, even though these methods have been used by many organizations and ISPs for some years, the academic community has provided only a limited evaluation of BGP Blackhole routing, using mainly network simulations.

The objective of this research was to evaluate the basic methods of BGP Blackhole routing in a real test-bed network in various environments. By using the response time, the CPU load, and the link load as performance metrics, we first evaluated the performance of those methods in networks where the routers’ CPU load was the limiting factor. Then we examined the effect of the high link load and the effect of routers’ preconfiguration on the BGP Blackhole routing’s performance.

The results showed that the BGP Blackhole routing may not be effective under stressful situations, that is, a high link load, because its dependence on TCP and the underlying routing protocols. Of the three basic Blackhole routing methods, the best method is the destination-based, followed closely by the source-based. The third method, customer-triggered Blackhole routing, in all cases had very degraded performance.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would like to thank my thesis advisor, Professor Geoffrey Xie. During this research I faced problems that seemed to me unsolvable. He always had solutions to those problems and more than often in less than five minutes. Especially in the end, I was very anxious, but his relaxed way of facing life helped me keep my heart rate below the maximum.

Special thanks to my thesis editor, Lee Rappold. She greatly improved my writing and her meticulousness in detail made my work easier.

I would also like to thank the Naval Postgraduate School, and especially Professor John Gibson, for providing the necessary equipment to create the test-bed networks.

Last but not least I would like to thank my military service branch, the Hellenic Air Force, for giving me the opportunity to visit this wonderful place and study in this institution. It was a dream for me since many years back.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

One of the most frequently used types of attack on computer networks is the Denial of Service (DoS) attack. A DoS attack’s main task is to stop a network device from providing specific legitimate services. This is usually accomplished by sending malformed traffic to the target or by sending a huge amount of normal traffic which will overload the target’s buffer. To be more effective, attackers often use many compromised machines, rather than just one, as a source for the attack. Thus the malicious packets approach the victim from different locations. This special type of DoS, called Distributed Denial of Service (DDoS), is one of the most difficult problems affecting normal operations on the Internet.

The economic implications of DDoS attacks are very significant. The FBI’s 2004 annual report on cyber crime, which is based on information provided by nearly five hundred organizations, found that a fifth of the victims that suffered financial losses had experienced DoS attacks. The total lost for the companies was over $26 million (Mirkovic and others 2005, 10). According to Anna Claiborne, a representative of Prolexic Technologies, in 2006 one major U.S. corporation lost over $2 million in a twenty-hour outage and an offshore gambling company lost an estimated $4 million in three days (Claiborne 2006, 18). During the last several years, there has been a constant demand for solutions to this serious problem and many different defense methods have been proposed.

One of the methods used by large network operators to block malicious DDoS packets is BGP Blackhole routing/filtering. Most modern routers have a special pseudo- interface, usually called Null0, which is always up and can never forward or receive traffic. Whenever a packet is pointed to Null0 it will be dropped, so this interface can be used to discard undesirable traffic. This method is more efficient than the use of Access Control Lists (ACLs), because it has no overhead and uses the highly optimized routing procedure of the router.

The Border Gateway Protocol (BGP) is the dominant protocol used for exterior routing between autonomous systems (AS) on the Internet. When BGP is running inside an AS, it is referred to as the internal BGP (iBGP). The BGP is very powerful and gives network administrators many options to apply as routing policies. Two routers that both speak BGP maintain a TCP connection in order to exchange information, such as routing advertisements.

The implementation of BGP Blackhole routing is relatively easy. In every router that is to apply Blackhole routing, a static route to the Null interface has been previously added. The static route uses a reserved, or private, IP address (e.g., 192.168.0.0/16). Another BGP-speaking router is used as a trigger. When a node inside the network is under attack, the trigger router is manually configured to advertise a new route. The advertisement says that, for the victim’s IP address, the next-hop is the reserved/private address noted above. But since this address points to the Null interface, the victim’s IP address also will point to the Null interface.

The main idea behind the use of BGP Blackhole routing is to block DDoS traffic as early as possible, which, in the case of an autonomous system, means applying it on border routers. These routers can handle more traffic, so they are more efficient in blocking a large amount of incoming packets. And since the traffic is blocked early, the AS is minimally affected by the attack. Since the target itself doesn’t receive any traffic, it is mostly still under DoS. Null routing does not discriminate between legitimate and malicious packets; it just drops everything that addressed the target. But since DDoS attacks also create collateral damage, BGP Blackhole routing mainly protects the neighbor network devices, that is, other nodes, routers, etc.

In keeping with the principals explained above, ISPs and router vendors have proposed a number of variations of BGP Blackhole routing. However, even though these methods have been used by many organizations and ISPs for some years, the academic community has provided only a limited evaluation of BGP Blackhole routing, using mainly network simulations. In his Master’s thesis, Kleffman evaluated the performance of BGP Blackhole routing on the U.S. Department of Defense (DoD) Non-Secure Internet Protocol Router Network (NIPRnet), using software simulation (Kleffman 2005, 107).

The objective of this thesis is to provide a scientific evaluation and analysis of the various BGP Blackhole routing methods, based on data collected on a real test-bed network. For this task, we first define the performance metrics. Then we evaluate the BGP Blackhole routing methods under different attack traffics and different limiting factors. Furthermore, we analyze the effect of the Blackhole preconfiguration on the normal operation of the routers.

In this research we assume that the DDoS attack has been positively identified by either automatic or manual means. However, the details of this identification are not part of this research. Though this topic is very important and needs more detailed research that can only be accomplished in a separate Master’s thesis.

This research concludes that, among the three basic BGP Blackhole routing methods, the destination-based method presents the best performance defined by the response time and routers’ CPU load. The second-best method is the source-based Blackhole routing. This method has a unique advantage among the three: it allows good traffic to reach the DDoS attack victim. But this aspect is not easy to implement in a real situation. Finally, the customer-triggered Blackhole routing in all cases had the worst performance.

The remainder of this thesis is organized as follows. Chapter II presents a more detailed explanation of DDoS attacks, BGP Blackhole routing methods, and the results of the most recent and detailed study of BGP Blackhole routing. Chapter III describes the methodology and the different test-bed network configurations we used in this research. Chapter IV presents the detailed results of this research. Chapter V provides conclusions and suggestions for future work.

II. BACKGROUND

This chapter provides background information for this study. The first section describes the basic aspects of DoS/DDoS attacks and the most common techniques that attackers use. The second section presents the principles of BGP Blackhole routing and the different methods of implementation that have been proposed to protect a network from DDoS attacks. The third section presents the results from previous studies related to BGP Blackhole routing analysis.

A. DOS/DDOS ATTACKS

The purpose of DoS/DDoS attacks is to disrupt the services offered by a host in a network. DoS attacks take advantage of various weaknesses in the IP protocol stack (Vayner 2003, 41). The two basic ideas behind DoS/DDoS attacks are 1) to send some kind of malformed traffic in order to create an abnormal operation of a victim’s network services, or 2) to send a large amount of traffic, usually normal, in order to consume a victim’s network resources.

The main distinctions between DoS and DDoS attacks are the source and the amount of the malicious traffic. In DoS attacks, the source is a single host, or a small group of hosts in the vicinity, and the traffic comes in much lower volume than in DDoS attacks. Since the source is limited, this kind of attack mostly exploits various software and design flaws (Vayner 2003, 41). A classic example of a DoS attack is the “Ping of Death.” An ICMP Echo request, or Ping, is usually 64 bytes in size. Most of the older computers could not handle a ping larger than 65,535 bytes, the maximum IP packet size. As a result, the reception of a packet that size or larger would create a system crash on the receiving host. By fixing the software flaws and applying the latest patches, potential targets can protect themselves from most DoS attacks. In recent years, therefore, that type of DoS attack has become very uncommon.

In contrast, in DDoS attacks, the attacker uses a very large number, usually hundreds, of compromised network systems that he has under his control to send a large amount of packets. In this case, the attacker exhausts the victim's limited resources, such as the bandwidth, router-processing capability, or network stack resources (Kleffman 2005, 107). The compromised systems, called "zombies" in computer science jargon, are usually network hosts in which the attacker was able to install special malicious software. The software gives the attacker the ability to control the host through network commands.

Very often, an attacker controls the "zombies" through other hosts, called Masters, and the communication might also be encrypted by the attacker to evade detection.

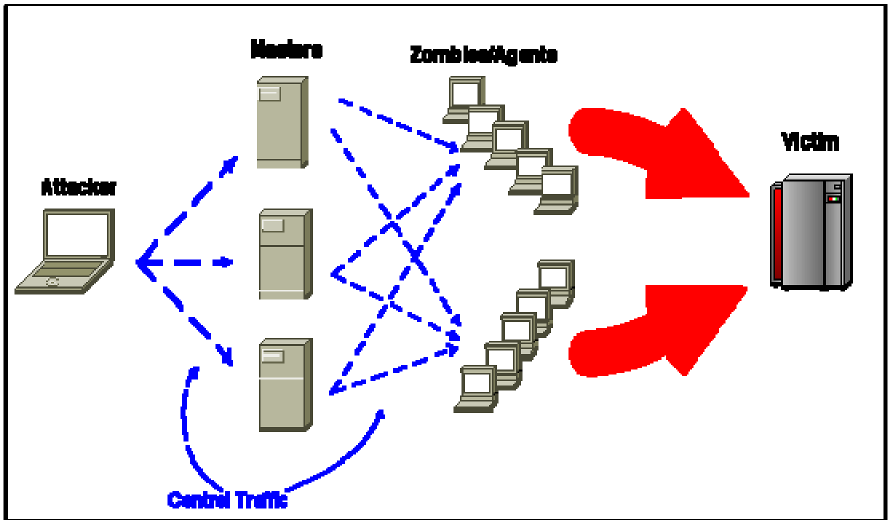


Figure 1. DDoS attack (From: Vayner 2003, 41)

A classic example of a DDoS attack is the SYN Flood attack designed to target servers. The attacker sends a large number of SYN packets through the compromised machines to a victim’s server. The packets all have a forged (“spoofed”) IP source address, so the server’s replies (SYN-ACK packets) will not be acknowledged with ACK packets. The victim’s queue will eventually fill up waiting for the ACK response from the attack machines and will not be able to service new TCP connections.

A DDoS attack can also be accomplished without zombies, by using legitimate network hosts. An example of this technique is the ICMP echo reply attack. The attacker sends a stream of ICMP echo requests to various hosts (usually by broadcasting), using the same “spoofed” source IP address that belongs to the victim. As a result, all replies go to the target creating a traffic storm.

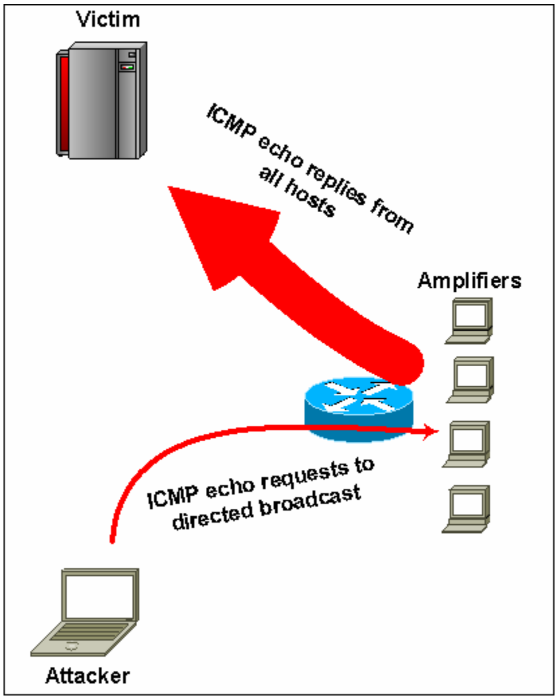

Victim
ICMP echo replies from all hosts
Amplifiers
ICMP echo requests to directed broadcast
Attacker

Figure 2. ICMP echo request attack (From: Vayner 2003, 41)

While DoS attacks have existed for decades, DDoS attacks are relatively new. The first documented cases appeared in late 1999. DDoS attacks became famous more recently after a series of attacks on popular e-commerce web sites like Yahoo!, Buy.com, Amazon, and eBay. Most of the companies lost their availability to the Internet for anywhere from a few hours up to a few days, causing hundreds of millions of dollars in lost revenue.

Because DDoS attacks have characteristics that differentiate them from other types of network attacks, these need to be taken into consideration when planning how to defend against them.

1. For a number of reasons, in DDoS attacks, it is very difficult to locate the attacker. The malicious traffic has a spoofed source IP address; it is not coming directly from the attacker's machine, but from hundreds of zombies. And between the zombies and the attacker are usually sited the "masters." Also, the communication between all players can be encrypted. Finally, the general absence of cooperation between the ISPs and network administrators makes it very difficult, if not impossible, to backtrack packets.

2. An attacker does not need to be technically sophisticated to launch a DDoS attack. Most of the tools for producing DDoS attacks are available for free on the Internet, and they are so simple that even teenagers can use them. In June 2001, for example, a web site, www.grc.com, became the target of a DDoS attack and was unavailable for many days. According to Steve Gibson, the web site owner, the attacker was a thirteen-year-old boy (Gibson 2005, 29). In February 2000, a series of DDoS

attacks to the popular sites eBay, Yahoo!, CNN, and many others was traced back to a fifteen-year-old boy from Canada (Vayner 2003, 41).

3. It is easy to create zombies. There are many public-access computers with Internet access that are poorly protected against an unauthorized installation of malicious code. Places like universities, colleges, public libraries, and Internet cafés usually have many of these potential zombies. PCs whose owners are unaware of the basics of computer security can also become zombies. This can happen, for example, if someone accepts emails or files from untrustworthy sources without previously scanning the files with anti-virus applications.

All the above characteristics make it difficult to defend against DDoS attacks. Existing DDoS defense techniques can be broken down into three main categories: prevention, detection, and response (Security Scape 2003, 3).

The first defense method, prevention, is all about stopping an attack before it starts. It requires the use of ingress and egress packet filtering based on the source IP address. The second method, detection, can be accomplished by a continuous monitoring of the network for patterns of attack. The patterns, such as abrupt changes in traffic for example, must be distinguished from normal network changes (Security Scape 2003, 3). After an attack has been identified, the final defensive action to be taken is a response. And the best response is to block, filter, or divert the malicious traffic away from the target. BGP Blackhole routing falls under this category of defense techniques. But a major problem with response mechanisms is that it is not easy to distinguish attack packets from normal ones. Therefore, more often than not, the only solution is to apply the same action to all traffic going to the victim.

B. BGP BLACKHOLE ROUTING

BGP Blackhole routing is one of the defense mechanisms used to block DDoS attacks. It combines a common feature of almost all modern routers, the Null0 interface, with the BGP routing protocol in order to drop packets that travel to a specific host.

Null0 is a pseudo-interface that every router has by default. It is always up but can never actually forward or receive traffic. Whenever a packet is routed to Null0, it will be dropped. Null0 works like the “/dev/null” directory in a UNIX Operating System. The main purpose of the interface is to discard undesirable traffic. Filtering through Null0 is a more efficient method than using Access Control Lists (ACLs), because it uses the highly optimized routing procedure of the router and thus incur much less processing overhead than ACL based packet filtering.

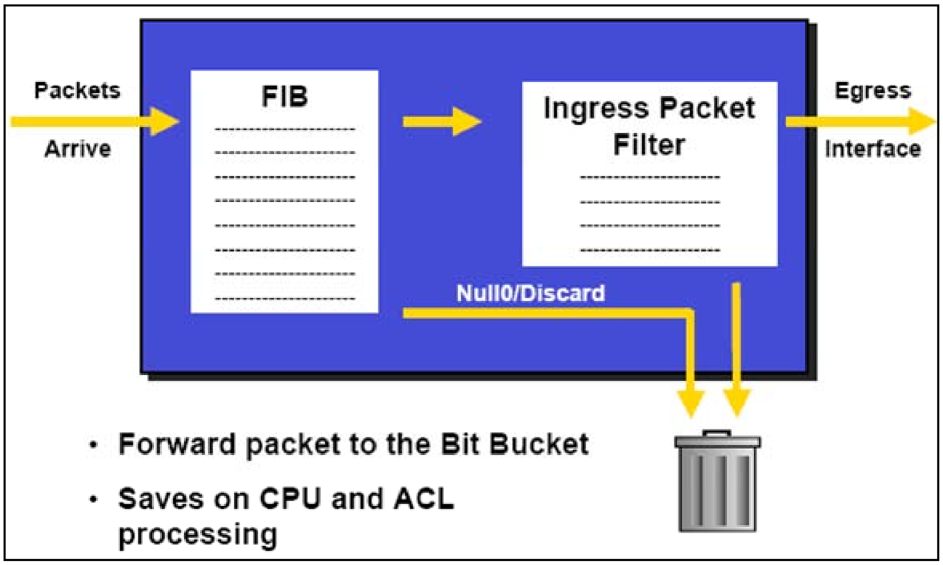
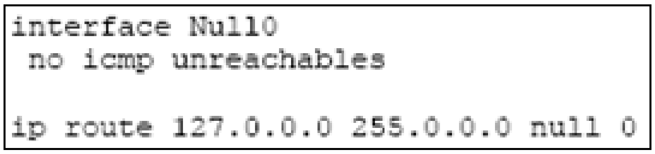


Figure 3. Use of Null0 interface (From: Battles, McPherson, and Morrow 2004, 47)

The configuration for applying Blackhole routing is simple. The basic requirement is a static route of the destination IP address to be discarded. Figure 4 shows this configuration for Cisco routers. When traffic is send to the Null0 interface, since there is no real host to receive the packets, ICMP Unreachable replies are generated by default. To prevent this unnecessary traffic, we use the first two lines from Figure 4. The lines first specify the interface and then configure the router to not create ICMP Unreachable replies for this interface.

The third line is the static route. In this example, the packets that have as their destination the subnet 127.0.0.0/8 will be forwarded to the Null0 interface.



```
interface Null0
 no icmp unreachables

ip route 127.0.0.0 255.0.0.0 null 0
```

Figure 4. Null0 routing (Cisco routers) (From: Raveendran Greene 2002, 10)

The Border Gateway Protocol (BGP) is the most popular routing protocol used between Autonomous Systems (AS). It is very powerful and gives network administrators many options in applying routing policies. When used inside an AS, it is called an internal BGP (iBGP). Routers that speak BGP establish a TCP connection between themselves, so that the exchange of information is reliable.

In BGP Blackhole routing, we want to block malicious traffic as early as possible. The best point to do so is the border routers. Usually, these are the most capable routing devices in a network, so they are the most efficient in handling large amount of packets. Furthermore, by discarding traffic at that point, we better protect our network, since no undesired traffic travels inside the AS.

The basic implementation of BGP Blackhole routing requires a preconfiguration of all border routers with a static route entry to the Null0 interface, using a private IP subnet address that is not used on the Internet, e.g., 192.0.0.0/24 (Battles, McPherson, and Morrow 2004, 47). A router inside the AS is also configured to work as trigger; it communicates with the border routers using iBGP. Although not specifically necessary, it is better to use a dedicated router for this purpose. It can be either a normal router or a workstation with software that can handle TCP/IP-based routing protocols like GNU Zebra or GateD (Battles, McPherson, and Morrow 2004, 47).

To apply Blackhole routing, a special static route to the IP address of the victim needs to be added to the trigger router. The static route contains more information under a “tag.” Among this information, the most important is the “next-hop,” which for Blackhole routing needs to be an IP address from the private subnet IP addresses already configured at the border routers. The trigger will automatically advertise the static route to the border routers, using an iBGP route update advertisement, and the border routers update their routing table with the new entry, forcing all traffic destined to the victim to be routed to the null interface. To stop Blackhole routing, simply remove the static route at the trigger router and the router will send out a route withdrawal to all border routers, again via iBGP.

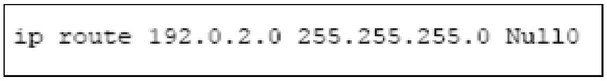
BGP Blackhole routing is not a perfect defense against DDoS attacks. Its most significant limitation is that it takes place on Layer 3 (Network Layer) and not on Layer 4 (Transport Layer) of the TCP/IP protocol stack. This means that the Blackhole routing technique blocks traffic based only on IP address. It cannot be more discrete in its filtering, for example, by dropping only telnet or HTTP packets going to the victim. Another drawback is that it is very hard to bypass or provide exceptions to the filtering, since to do so we have to actually bypass the router’s forwarding table (Raveendran Greene 2002, 10).

In the last few years, many variations to the basic Blackhole routing technique have been proposed, all of which can be categorized as one of two basic implementations: the Remote-Triggered (RTBH) and the Customer-Triggered. The main distinction between the two is the origination of the filtering command. RTBH routing can be further divided into either destination-based routing or source-based routing, depending on what information (the source or the destination IP address) is used to block traffic.

1. Remote-Triggered Blackhole Routing

a. Destination-Based

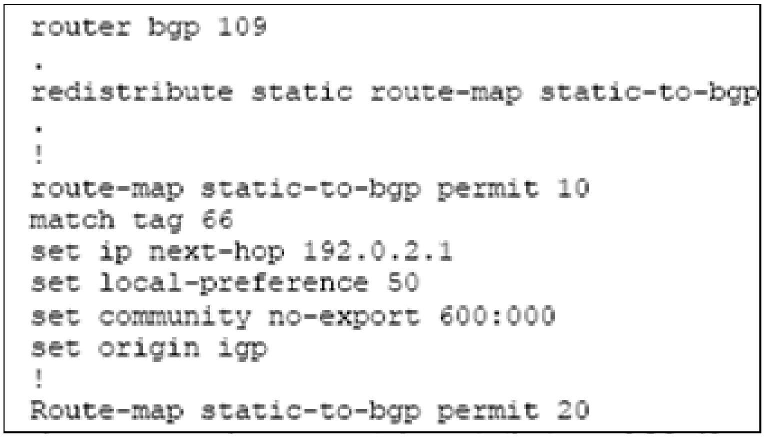
RTBH routing is the basic version of BGP Blackhole routing. The border routers need a static route to the Null0 interface using a test subnet IP address that is not used on the Internet. Figure 5 shows the configuration for a Cisco border router. As noted above, the use of the "no icmp unreachables" command is optional, but highly recommended.



```
ip route 192.0.2.0 255.255.255.0 Null0
```

Figure 5. Border router set-up (From: Raveendran Greene 2002, 10)

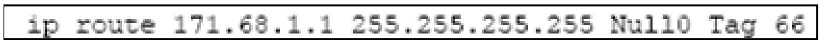
The trigger router is installed at the Network Operations Center (NOC) and is better if dedicated for this purpose and to accept no routes. The basic configuration of the trigger router is shown in Figure 6. In that example, the router belongs to AS 109, the name of the static route-map is "static-to-bgp", and the "community no-export" command is used to deny advertisement of the Blackhole routing outside the AS.



```
router bgp 109
.
redistribute static route-map static-to-bgp
.
!
route-map static-to-bgp permit 10
match tag 66
set ip next-hop 192.0.2.1
set local-preference 50
set community no-export 600:000
set origin igp
!
Route-map static-to-bgp permit 20
```

Figure 6. Trigger router set-up (From: Raveendran Greene 2002, 10)

The trigger router must talk iBGP to every border router. To activate the RTBH routing, we add a new static route to the trigger, using the same tag value as defined above. Figure 7 shows the command issued to the trigger router. In that example, the IP address 171.168.1.1/32 belongs to the target to be protected.



```
ip route 171.68.1.1 255.255.255.255 Null0 Tag 66
```

Figure 7. Activation of trigger router (From: Raveendran Greene 2002, 10)

When the static route reaches the border routers, their routing table is updated by the procedure shown in Figure 8.

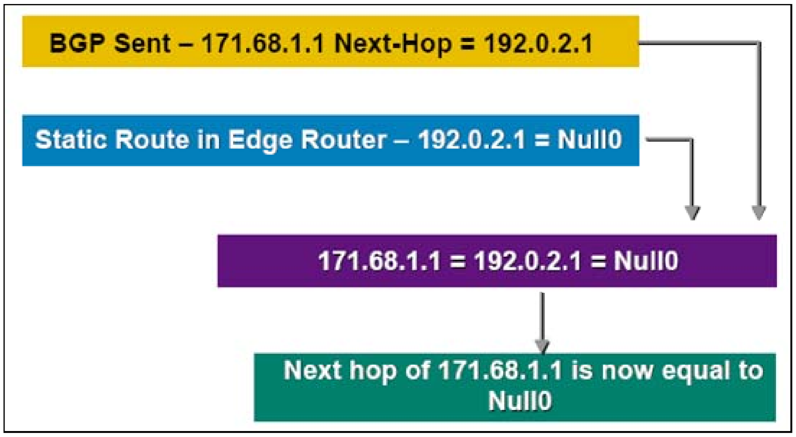

BGP Sent – 171.68.1.1 Next-Hop = 192.0.2.1
Static Route in Edge Router – 192.0.2.1 = Null0
171.68.1.1 = 192.0.2.1 = Null0
Next hop of 171.68.1.1 is now equal to Null0

Figure 8. Activation of Blackhole routing (From: Raveendran Greene 2002, 10)

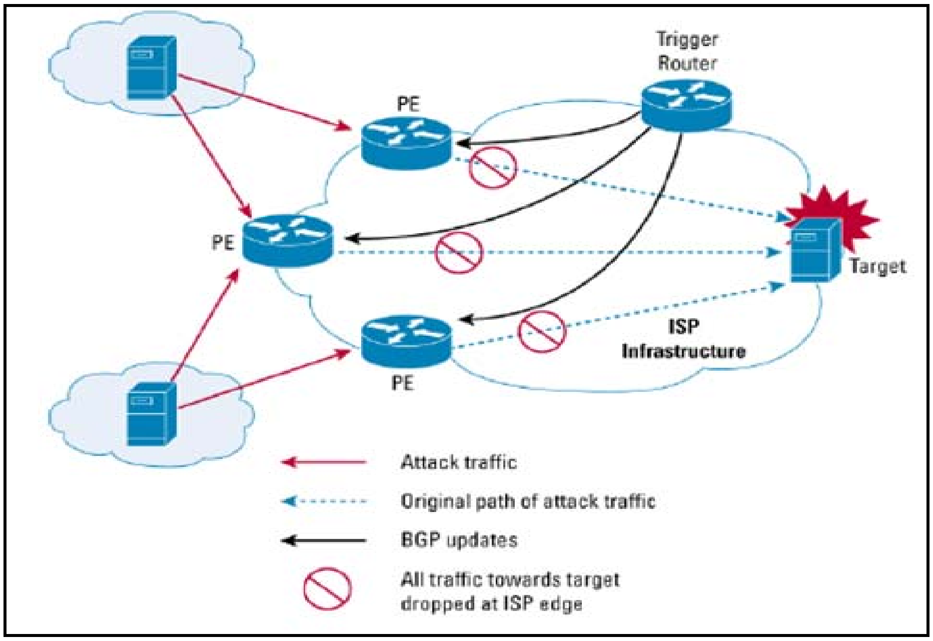

Trigger Router
PE
PE
PE
Target
ISP Infrastructure
Attack traffic
Original path of attack traffic
BGP updates
All traffic towards target dropped at ISP edge

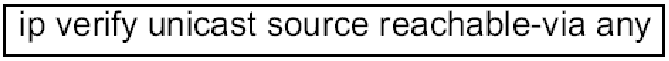
Figure 9. Destination-Based RTBH routing (From: Cisco 2005, 37)

To disable the Blackhole routing, a new command must be issued to the trigger router, which removes the static route; the trigger router then sends out an iBGP withdrawal.

b. Source-Based

Source-based RTBH routing provides the ability to drop traffic based on either a specific source IP address or a range of source addresses (Cisco 2005, 37). This method allows legitimate traffic to reach the victim of the DDoS attack. Implementation of source-based RTBH routing depends on unicast Reverse Path Forwarding (uRPF), more often, loose mode uRPF. The router checks the source IP address of the packet and if there is no entry in the Forwarding Information Base (FIB) for the specific address, or if the entry points to Null0, the packet is dropped

(Cisco 2005, 37). Figure 10 shows the command that has to be added to a Cisco router configuration file for every interface that needs to use uRPF.



```
ip verify unicast source reachable-via any
```

Figure 10. uRPF command for Cisco routers

The configuration of the border and trigger routers remains the same as in the destination-based method. The only difference is that, for the source-based routing, the uRPF must be configured on all external facing interfaces of the border routers. To activate the Blackhole routing, a static route must be added to the trigger router; but this time, the IP address (or range of addresses) of the attacker is used. The withdrawal procedure is the same as in the destination-based method.

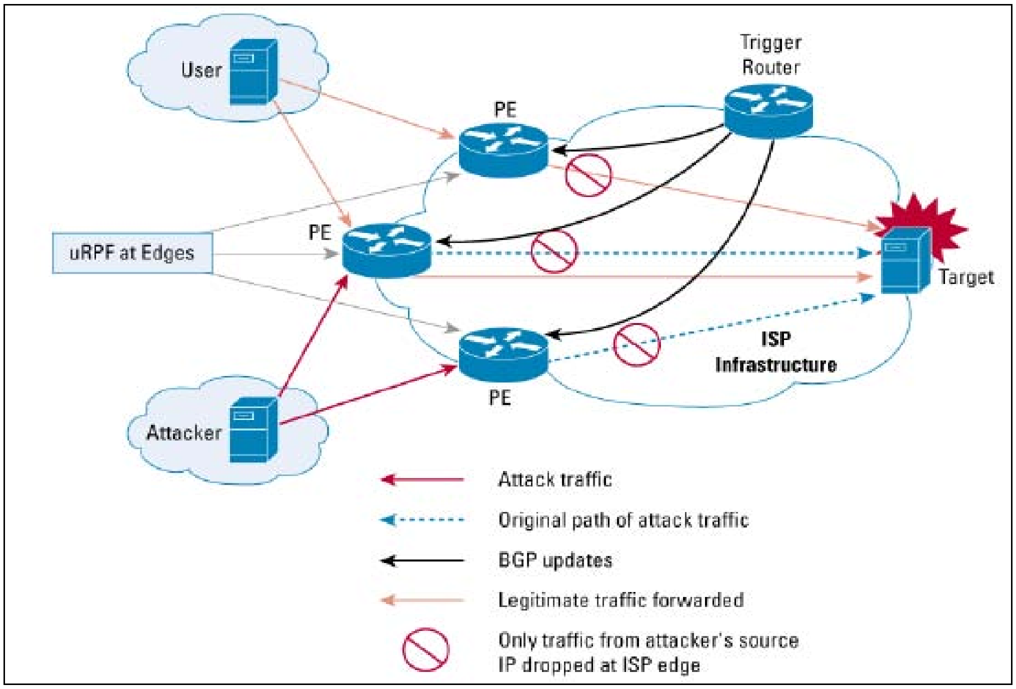

User
Trigger Router
PE
PE
uRPF at Edges
Target
ISP Infrastructure
PE
Attacker
Attack traffic
Original path of attack traffic
BGP updates
Legitimate traffic forwarded
Only traffic from attacker's source IP dropped at ISP edge

Figure 11. Source-Based RTBH routing (From: Cisco 2005, 37)

2. Customer-Triggered Blackhole Routing

The main difference in customer-triggered Blackhole routing, as compared to RTBH routing, is that the activation does not come from a trigger router controlled by the ISP or AS administrator, but from a customer-owned device.

This technique requires a pre-agreement between the ISP and the customer, because the latter needs information that is not usually available from ISPs. Thus, the ISP needs to properly configure the border routers to accept the customer’s iBGP advertisements.

However, even though this method gives customers the ability to respond faster to a DDoS attack, the fact that the advertisements must pass through the same links as the DDoS attack is an indication of a potentially slower response. This was demonstrated by M. Kleffman’s thesis at the U.S. Air Force Institute of Technology (AFIT) (Kleffman 2005, 107). It is illustrated by the analysis done in the present study.

C. PREVIOUS STUDY OF BGP BLACKHOLE ROUTING

As noted in the introduction, Blackhole routing has been used by ISPs and other network administrators for several years, but there are only a limited number of evaluations of the technique. The only detailed analysis is M. Kleffman’s thesis (Kleffman 2005, 107), which analyses the performance of BGP Blackhole routing on a network, such as the Non-secure Internet Protocol Router Network (NIPRNET) used by the U.S. Department of Defense (DoD).

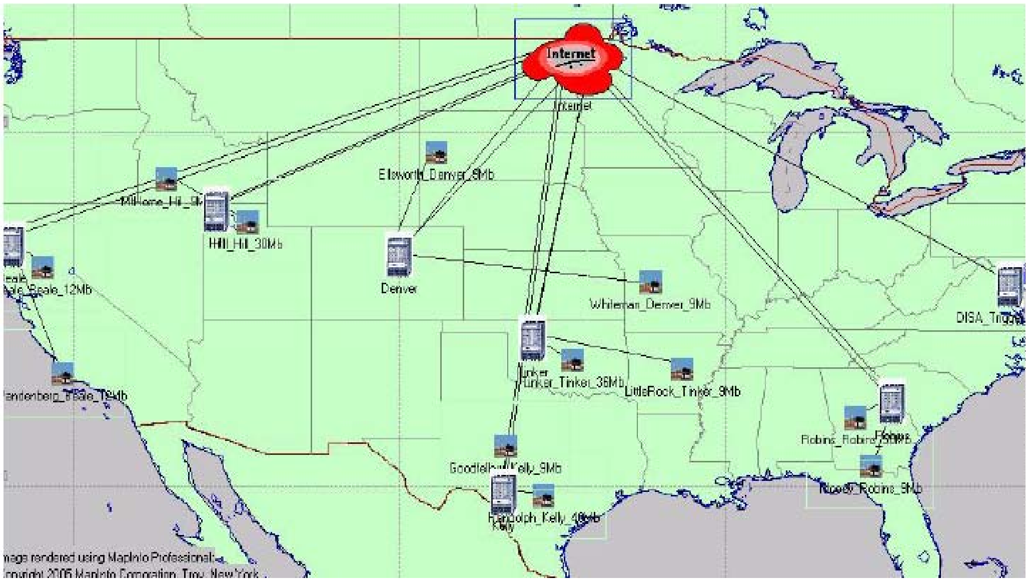


Figure 12. Simulated NIPRNET network, printed from OpNet Modeler application (From: Kleffman 2005, 107)

NIPRNET is a virtual network connecting smaller networks around the world. Each of the smaller networks is connected to a border router owned by the Defense Information Systems Agency (DISA) that is connected to the Internet. The border routers are also connected to each other via a Virtual Private Network (VPN). Even though the study focuses mainly on NIPRENET, some of the results are applicable generally to traditional networks like the Internet.

1. Goals and Methodology

Kleffman’s main goals were to: determine if BGP Blackhole routing was effective in defending NIPRNET against DDoS attacks; determine the effectiveness of Blackhole routing with one or more border routers out of the defense mechanism; and determine the effectiveness of customer-triggered against remote-triggered Blackhole routing.

The evaluation method Kleffman used was a simulation based on OpNet Modeler version 10.5 (Kleffman 2005, 107), in which the simulated network consisted of six border routers, a trigger router, and twelve customer routers. The customers were equivalent USAF Bases. The bandwidth of the communication links varied from 9 Mbps up to 40 Mbps.

Four different workloads were defined. The first was normal operations traffic based on real data obtained from the Air Force Network Operations Center (AFNOC). This was the baseline workload. The other three workloads consisted of the baseline workload plus DDoS attack traffic of 12.8 Mbps, 38.4 Mbps, and 64 Mbps, respectively, which originated from six attack systems (Kleffman 2005, 107). In all the scenarios a non-response period of ten seconds between the DDoS attacks' initiation and the Blackhole routing advertisements simulated the IDS response time. A total of 39 different experiments were executed.

The performance metrics chosen were:

1. Queuing delays on each router,
2. Latency between the bases (customer routers) and the border routers and between the trigger router and the border routers,
3. Router convergence delay, and
4. Bandwidth utilization on the links between the bases and the border routers.

2. Results

a. Effectiveness of BGP Blackhole Routing on NIPRNET

The first result demonstrated the effectiveness of BGP Blackhole routing. Initially, the utilization and the latency inside the NIPRNET were measured with the baseline traffic. Then, after applying the Blackhole routing, they were measured with the three different DDoS attack scenarios. The simulation showed that, inside the NIPRNET, the utilization and the latency remained the same.

Next, the queuing delay of the border routers was measured. The results showed that, with increasing traffic, the respective queuing-delay increases were very small. This suggests that the impact the BGP Blackhole routing had on the queuing delays of the border routers was related to the increase in attack traffic, not to the actual dropping of the packets (Kleffman 2005, 107).

The measurements of the utilization, latency, and queuing delay all confirm that the BGP Blackhole routing was effective in protecting the NIPRNET.

b. Effectiveness of BGP Blackhole Routing without All the Border Routers Participating

The next step was the evaluation of BGP Blackhole Routing when only some of the border routers were participating in the filtering. For this measurement, four different scenarios were simulated. In the first scenario, the NIPRNET was defended by only the border router directly connected to the target base. In the next three scenarios, a combination of one, three, and five out of a total of six border routers (not directly connected to the target) was chosen.

The first metric analyzed was the inbound bandwidth utilization. The measurements showed that utilization increased up to 100 percent as the number of defending routers decreased. The only exception was when the border router directly connected to the target participated in the filtering procedure: in that case, the utilization didn’t change. But that result occurred because the NIPRNET is not a traditional AS, but rather a VPN connection of smaller networks spread around the world that are not directly connected to one another.

Next, the queuing delay on the border routers and the inbound latency of the communication links under attack were measured for the same scenarios. The simulation showed that the queuing delay increased, but in such a small percentage that it should not have a major impact on the performance of the network (Kleffman 2005, 107).

The latency also increased, both because of the queuing delay and, in some cases, because of the link saturation. The latter happened when the DDoS attack workload was much higher than the bandwidth of the link and the directly connected border router didn’t participate in the filtering.

The simulation’s least surprising result was that the Blackhole routing was not as effective when only some of the border routers participated in the filtering. Furthermore, the bandwidth of the communication link under attack plays an important role, especially as the attack traffic becomes larger (Kleffman 2005, 107).

c. Comparison of Remote-Triggered with Customer-Triggered Blackhole Routing on NIPRNET

For the Remote-Triggered Blackhole routing, the topology remained the same as that discussed in the previous sections. For the Customer-Triggered Blackhole routing, the updating of the border routers started from the router of the base under attack. First, the study measured the convergence time on the routers, which, in the case of Customer-Triggered Blackhole routing, showed a significant increase in the values of that metric. Although in the RTBH filtering, the maximum convergence never exceeded the 32 milliseconds, in the second method, the maximum convergence boosted to more than 45 seconds. That result by itself is enough to show the advantage of RTBH routing.

The study also compared the bandwidth utilization, the queuing delay on border routers, and the latency. All three metrics show increased values with the Customer-Triggered Blackhole routing methodology, especially in high-volume attacks. In the case of a 64-Mbps attack, the queuing delay elevated from microseconds to seconds (Kleffman 2005, 107).

3. Comments

This paper shows that BGP Blackhole routing is effective as a defense mechanism against DDoS attacks on the NIPRNET. And since the Internet and traditional networks have the same basic principles as the NIPRNET, the effectiveness of Blackhole routing would apply also to them, perhaps with even better results.

While Kleffman’s thesis provided very useful results, it was based on a simulation. Simulation is an excellent analysis tool, especially in cases in which the real thing cannot be tested or is very difficult to be created. But simulations also have some disadvantages. The most important drawback is that they are based on simplifications and assumptions. Theoretically, depending on how good the simulation is, the simplifications do not affect results. But in order to accept the data provided by a simulation, there must be a minimum base of experimental data that points to the same results. One of the main objectives of our thesis is to provide that experimental data and either verify or call into question the conclusions of Kleffman’s thesis.

Another shortcoming of simulations is that they usually do not show unexpected or hidden results based on the irregular behavior of the test object. For example, when some routers work under extreme traffic, they might present unstable behavior that affects their performance but that can only be shown in real experiments.

Simulations are very demanding applications, as they are highly dependent on the performance of the computer within which they are installed. Furthermore, OpNet is a discrete event simulation and as such it uses event times based on calculated or expected delays which can present variations from the real values.

To do a more complete analysis of BGP Blackhole routing, measurements on a real test-bed network must be taken and then be compared to those of the simulation. Our thesis tries, in part, to address that issue.

Since the router is the main player in BGP Blackhole routing, its performance is a very important factor and must be analyzed. Kleffman’s study does not address the issue. Here, we analyze how the router performance, in terms of the CPU load, affects the Blackhole routing. This metric is more generic and thus more useful. Furthermore, we analyze the effect of link load on Blackhole routing. Depending on the devices used in a network, the limiting factors (i.e., the CPU load and the link load) can produce varying results. Therefore, we examine both factors.

One of the main shortcomings of BGP Blackhole routing is the absence of an automation initialization procedure. At present, after identifying an attack, the network administrator has to manually add the static route to the trigger router. This obviously increases the response time. In every one of its simulation scenarios, the AFIT study assumes that there is an IDS/IPS that identifies attacks and, after ten seconds, initiates the Blackhole routing. But the study does not explain how this can actually be implemented. That issue, if solved, would greatly increase the effectiveness of Blackhole routing. For this reason, we investigated possible solutions to the automation problem.

As noted, Kleffman’s thesis focuses mainly on the NIPRNET, though some of the results may also apply to more generic networks like the Internet. But that possibility must be evaluated and confirmed. In our study, therefore, we use a more generic real test-bed network in order to verify the AFIT study results and also produce additional results that give a more detailed picture of the BGP Blackhole routing methodology. In the next chapter, we present a detailed description of our methodology and the test-bed network we used.

THIS PAGE INTENTIONALLY LEFT BLANK

III. GOALS AND METHODOLOGY

This chapter describes the goals of our research and the methodology we used to accomplish them. As noted previously, the methodology that best fit our goals was data collection and analysis from real test-bed networks. The details of these networks, starting from their basic configuration up to the data collection, are all explained in the following pages.

A. GOALS

This research had four goals:

1. Describe the proposed methods of BGP Blackhole routing.
2. Define performance metrics that can be used to evaluate the BGP Blackhole routing methods.
3. Evaluate those methods using the performance metrics defined above.
4. Analyze the effect of pre-configuration on routers normal operations.

In the previous chapter, the proposed methods of BGP Blackhole routing were described in detail. In this chapter, the performance metrics and the test-bed networks we used are described. The final results of this research are presented in Chapter IV.

Before continuing, we must clarify an important aspect of our research. Though the identification of a DDoS attack is closely related to our work, it is not part of this thesis. During the data collection from the different test-bed networks that we created, we assumed that the DDoS attack had been positively identified by other means, i.e., by either an IDS/IPS or a human closely monitoring the traffic inside the network. In either case, the attack information is passed to the proper authority, which, in our case, is the network administrator.

The reason we do not include this topic in our thesis is that, because it is such an important and difficult issue, it merits a dedicated study in its own right.

B. PERFORMANCE METRICS

The performance metrics chosen for this research were: the router response time, the router CPU load, and the link load.

The router response time is the time between the moment when the static route to a victim’s IP address is entered manually or automatically in the trigger router and the moment when the border routers update their routing tables with the new static route entry. DDoS attacks can be initiated almost instantaneously and with no indication on the victim’s side of an incoming attack. In other cases (Gibson 2005, 29), the attack may gradually increase. In any case, the network must be able to respond quickly. The router response time is the most important performance metric, because it shows how fast a network can be protected from a DDoS attack.

As noted in the previous chapter, Kleffman’s thesis shows, the effectiveness of BGP Blackhole routing is degraded when not all border routers are participating (Kleffman 2005, 107). In light of that factor, in our research we assumed that all the border routers were part of the Blackhole routing defense mechanism, and we did not examine the issue any further.

In the process of our initial tests with simulated DDoS attacks, we realized that there are two factors that affect the performance of BGP Blackhole routing: the router CPU load and the link load.

The router CPU load is a limiting factor when the malicious traffic is much higher in volume than the router’s forwarding capability. When the routers are more capable, the links inside the network usually become congested, and as such, are a limiting factor. In our research we investigated the performance of BGP Blackhole routing in both situations.

Since all proposed BGP Blackhole routing methods require preconfiguration in both the trigger router and the border router, we analyzed the effect of this preconfiguration on routers’ normal operation. As a metric for the comparison, we used the routers’ CPU load.

C. NETWORK CONFIGURATION

The main methodology used for this research included the simulation of DDoS attack scenarios in different real test-bed networks and data collection of performance metrics with the use of proper tools. This was accomplished in three steps. The first step was to set up appropriate test-bed networks that would simulate real DDoS attacks and BGP Blackhole routing scenarios. The second step was to use appropriate hardware and software to create malicious traffic. The third and final step was to use appropriate applications and methods to collect the data defined by the performance metrics, to analyze the data, and to produce conclusions.

1. Hardware and Software

For the test-bed networks we used the following devices:

1. Five Cisco 2621XM routers with IOS 12.3(15b). Each router had 32Mb of Flash memory, 32Mb of RAM memory, four 10Mbps Ethernet interfaces, and two 100Mbps Fast Ethernet interfaces.
2. One Cisco 3600 router with IOS 12.2(5d), 16Mb of Flash memory, 16Mb of RAM memory, eight 10Mbps Ethernet interfaces, and one 100Mbps Fast Ethernet interface.
3. One Juniper J4300 router with JUNOS 7.1R1.3, 256MB of Flash memory, 256MB of RAM memory, and six 100Mbps Fast Ethernet interfaces.
4. One Cisco Catalyst 1900 switch with twenty-four 10Mbps interfaces.
5. One D-Link DGS-1004T switch with four gigabit interfaces.
6. One SmartBits 6000C Performance Analysis System of Spirent Communications for use as a traffic generator.
7. One LAN-3321A TeraMetrics XD module with two 10/100/1000 Mbps Ethernet Copper ports and two Gigabit Ethernet Fiber ports installed on the SmartBits 6000C system.
8. One desktop PC, acting as the target.
9. One laptop PC, acting as the network analyzer, installed with all the applications noted below.

The applications we used during the various tests are the following:

1. SmartWindow version 7.70.128, for use with the SmartBits 6000C system.
2. SolarWinds Standard Edition version 8.2, for network management and analysis.
3. CommView version 5.1 of Tamosoft, for creating custom packets.

4. Kiwi Syslog Deamon version 8.0.2 of Kiwi Enterprises, for capturing SNMP messages.

5. Ethereal version 0.10.14, for capturing normal traffic.

2. Test-Bed Networks

To evaluate the performance of the BGP Blackhole routing methods, we created three different test-bed networks. The first test-bed network was used to measure the performance of the two basic remote-triggered BGP Blackhole routing methods: destination-based filtering and source-based filtering. This network was also used to analyze the effect of the router CPU load as a limiting factor inside the network.

The second test-bed network was used to measure the performance of the customer-triggered Blackhole routing. The effect of the router CPU load was also measured.

The third test-bed network was used to measure the effectiveness of BGP Blackhole routing when the limiting factor was the link load.

a. Test-Bed Network #1

As noted above, the main task of test-bed network #1 was to evaluate the performance of destination-based and source-based RTBH routing. The main idea was to create a DDoS attack from the packet generator that would target a host inside an Autonomous System (AS). The malicious traffic would approach the AS from different sources and thus had to pass through different border routers. After the initialization of the attack, the trigger router inside the AS was configured to advertise either destination- or source-based Blackhole routing. Using analysis tools, we then measured the response time of each border router under different amounts of attack traffic. Using the same tools, we also measured the router CPU load under different volumes of attack traffic and analyzed the effect it had on the average response time.

Using the available hardware, we simulated an AS with three border routers, two internal routers, one trigger router, and one desktop PC as the target of the attack. All the routers were the Cisco 2621XM model except the one closest to the target, which was a Cisco 3600 model. One laptop PC with the analysis tools was connected to the latter router. The three border routers were connected to the SmartBits packet generator’s two interfaces, either directly or through a Cisco 1900 switch. All interfaces in the network were 10Mbps Ethernet. The test-bed network is presented in Figure 13.

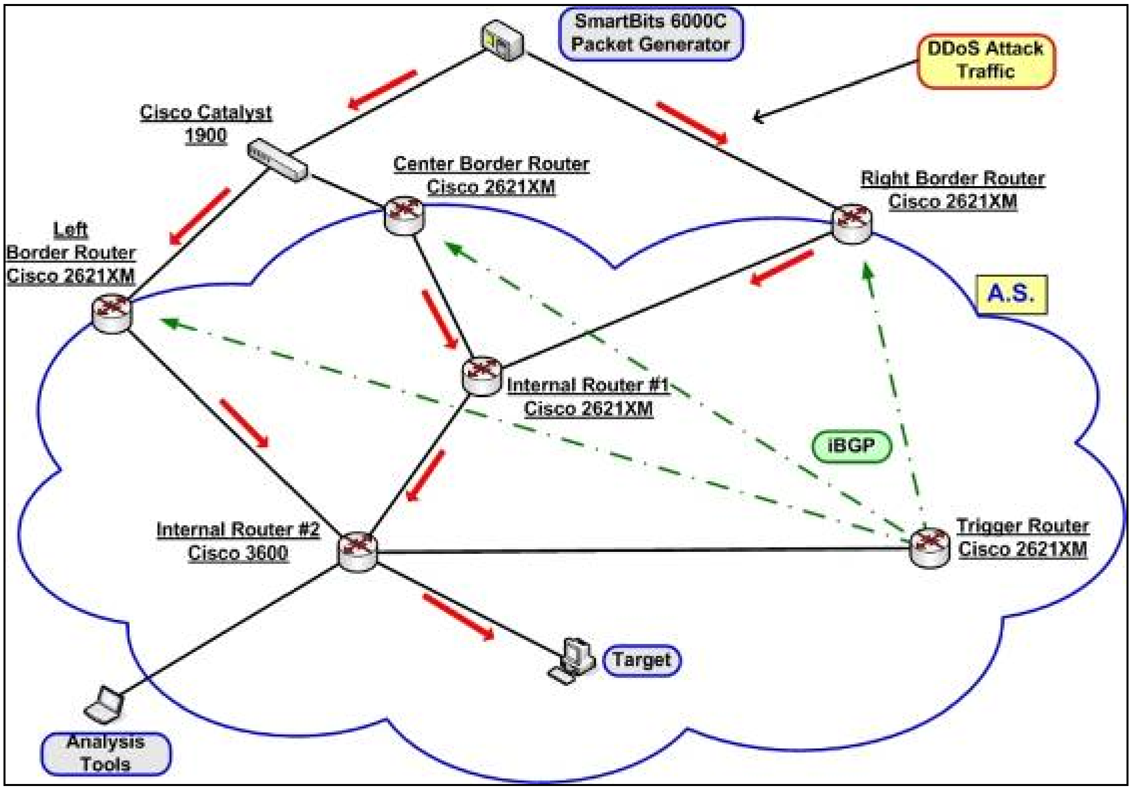


Figure 13. Test-bed network #1

The main routing protocol used inside the network was OSPF. To advertise the Blackhole routing, the trigger router had iBGP peer-to-peer connections with the border routers.

With the obvious exception of different IP-address assignments for the interfaces, the configuration of the three border routers was the same. The same was true for the two internal routers. The actual configuration files of one internal router, one border router, and the trigger router for test-bed network #1 are presented in Appendix A.

b. Test-Bed Network #2

The main purpose of test-bed network #2 was to evaluate the customer-triggered Blackhole routing and then compare it to the remote-triggered Blackhole routing. The topology remained almost the same as that of test-bed network #1; the only difference was the position of the trigger router. To simulate a customer network inside the AS, we positioned the trigger router in line with the target host. The trigger router in this case thus also played the role of the border router for the small customer’s network.

The details of the proper authorization for the customer to communicate with the AS border routers did not affect our evaluation; that is more an issue of agreement between either the ISP or the AS network administrator and the customer.

The topology of test-bed network #2 is presented in Figure 14.

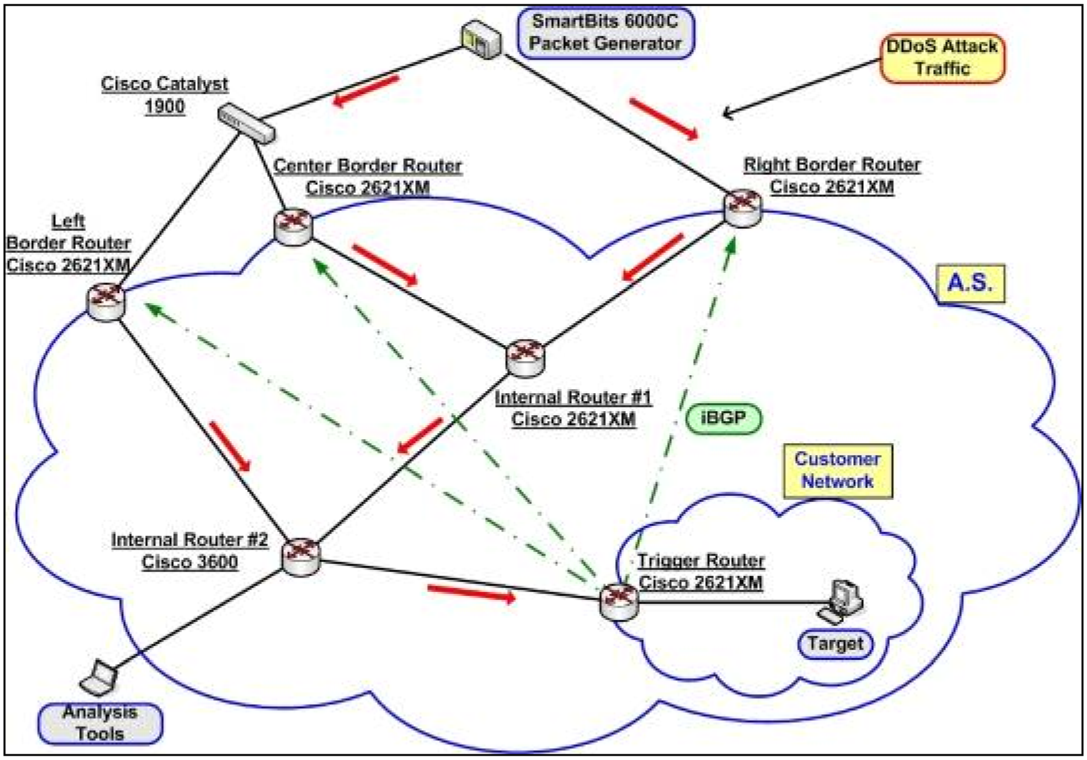


Figure 14. Test-bed network #2

The configuration file of the trigger router in test-bed network #2 is presented in Appendix B.

c. Test-Bed Network #3

The purpose of test-bed network #3 was to evaluate the performance of BGP Blackhole routing in a network where the routers have sufficient CPU capacity, so the limiting factor could be the link load, not the router CPU load. For this network we used four Cisco 2621XM models (three border routers and a trigger router), one Juniper J4300, one D-Link Gigabit switch, a desktop PC as the target, and a laptop PC with the analysis tools. The topology of the network is presented in Figure 15.

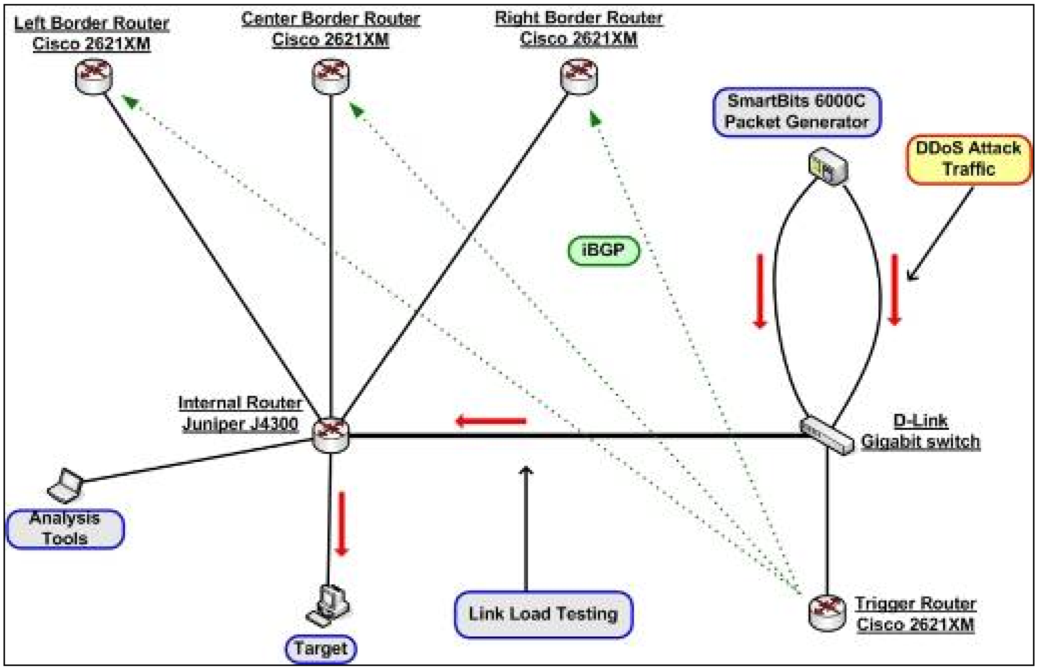


Figure 15. Test-bed network #3

The links target-J4300 and the switch-SmartBits were configured to 100 Mbps; the link J4300-switch was configured to 10Mbps. All the routers used OSPF as the main routing protocol; iBGP sessions were also established between the border routers and the trigger.

Since both the Juniper J4300 router (unlike the Cisco 3600) and the gigabit switch were able to handle much more than the 10Mbps traffic, by downgrading the link J4300-switch to 10Mbps we created a bottleneck link inside the network. And thus the link load became the limiting factor.

The main idea for this experiment was to load the link J4300-switch with different traffic loads and then apply Blackhole routing. The iBGP updates of the trigger router to the border routers had to pass through the loaded link. Using the tools and methodology explained in subsequent pages, we measured the effect of the link load on the three border-routers’ response times. In this network the BGP Blackhole routing didn’t actually block the malicious traffic, but this was not the objective of the evaluation.

The basic configuration of the trigger router and the border routers were the same as in the previous test-bed networks. The configuration of the Juniper J4300 router is presented in Appendix C.

D. TRAFFIC GENERATION

To properly evaluate the various BGP Blackhole routing methods, we had to create DDoS attacks for all of the test-bed networks described above. The hardware available for this task was the SmartBits 6000C Performance Analysis System of Spirent Communications, with one LAN-3321A TeraMetrics XD module with two 10/100/1000 Mbps Ethernet Copper ports and two Gigabit Ethernet Fiber ports. The device is capable of creating customized layer-three and layer-four packets in IPv4 and IPv6 formats. Furthermore, it allows users to customize layer-two information (i.e., source and destination MAC address). All the ports of the module can send and receive traffic simultaneously (full duplex), and the interfaces can act as regular hosts inside a network. To control the system, we used the SmartWindow version 7.70.128 Graphical User Interface (GUI) application. Figure 16 shows the main screen of this application.

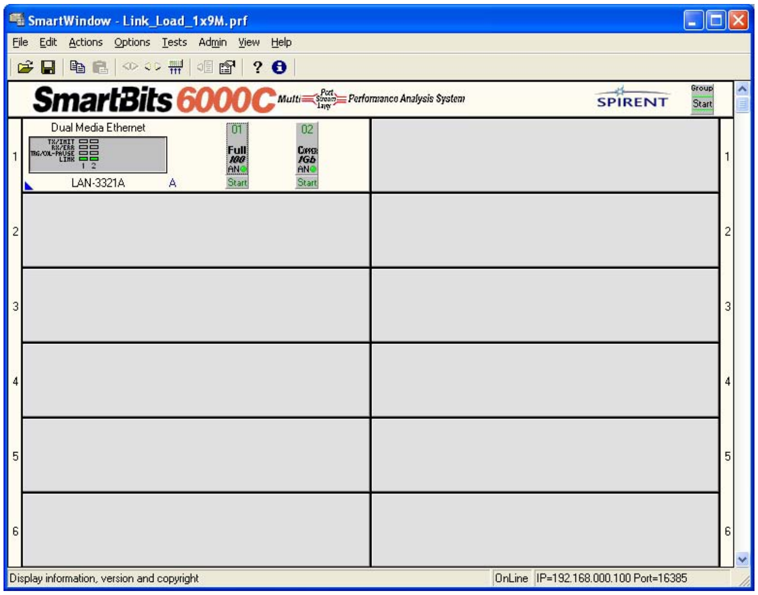
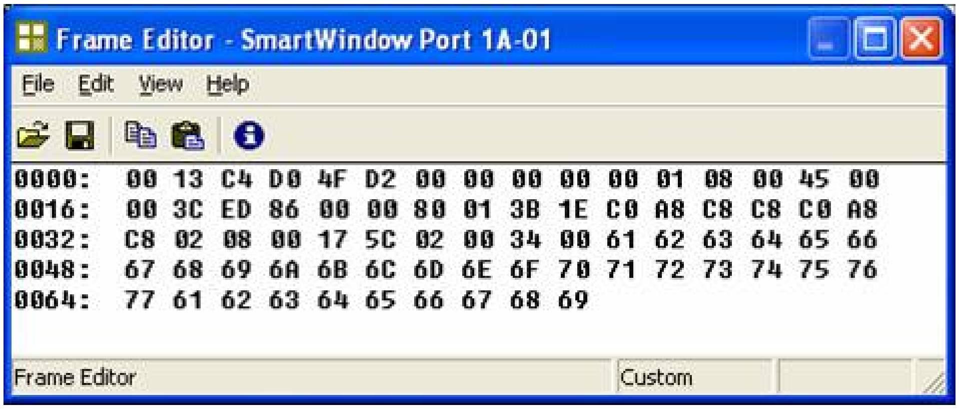

SmartWindow - Link_Load_1x9M.prf
File Edit Actions Options Tests Admin View Help
SmartBits 6000C Multi Port Stream Layer Performance Analysis System
SPIRENT
Group
Start
Dual Media Ethernet
LAN-3321A
A
01
Full
100
AN
Start
02
Copper
1Gb
AN
Start
1
2
3
4
5
6
Display information, version and copyright
OnLine
IP=192.168.000.100 Port=16385

Figure 16. SmartWindow GUI

In the initial configuration of the test-bed networks we also used the CommView 5.1 network monitor and analyzer application of Tamosoft. The main features of this product that were most valuable to us were the packet generator and traffic capture capability. The application didn’t have the capability to send the amount of traffic we needed, but the user-friendly interface helped us create the traffic that was most efficient for our purposes in a very short time.

With CommView we created two custom ICMP Echo-Request packets to be used by the packet generator for the DDoS attack. In both cases, the destination IP address was that of the target, but the source IP address was either the same as the destination or one that wasn’t assigned to any host inside the test-bed network and for which only the last bit of the IP address was different from the destination IP address. The use of these two specific fake-source IP addresses prevented the ICMP Echo-Replies from traveling inside the network and thus altering the measurements. For the initial task of this network, only one of the custom packets was needed. But after our first experiments we observed that, depending on the source IP address of the packet, the filtering on the border routers resulted in different CPU loads. So, to further investigate this phenomenon we used two custom packets. A hexadecimal form of one of the packets is presented in Figure 17.



```
0000:  00 13 C4 D0 4F D2 00 00 00 00 00 01 08 00 45 00
0016:  00 3C ED 86 00 00 80 01 3B 1E C0 A8 C8 C8 C0 A8
0032:  C8 02 08 00 17 5C 02 00 34 00 61 62 63 64 65 66
0048:  67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76
0064:  77 61 62 63 64 65 66 67 68 69
```

Figure 17. Custom ICMP Echo-Request packet

Using the packets described above, we then created the attack flows with the SmartBits application. Thirteen attack flows were defined, with different amounts of traffic, starting from a small number of frames per second (fps) up to the maximum capability of the packet generator for the specific connections we created. Each flow was divided equally into three parts, so that only a third of the total traffic would pass through a single border router. For the final flow, we used the maximum bit- rate of the packet generator in every port, thus the right-side border router had to handle an amount of traffic that was equal to the traffic of both the other border routers combined. These flows are presented in Table 1.

Flow #	Total Frame Rate (fps)	Total Bit Rate (Mbps)
1	1500	0.89
2	3000	1.77
3	4500	2.66
4	6000	3.54
5	7500	4.43
6	9000	5.32
7	10500	6.20
8	12000	7.09
9	12780	7.55
*10	12900	7.63
*11	16500	9.74
*12	19134	11.33
*13	25511	15.10

Table 1. Attack Flows

Flow #9 was the maximum flow under which the CPU load of every router was below 80 percent in test-bed network #1. Above this level some of the routers reached a maximum CPU load and their behavior became very unstable. Because of that, we assume that Flows #10 through #13 simulate more accurately a DDoS attack in networks where the limiting factor is the router CPU load. All thirteen flows were used in both test-bed network #1 and #2.

For test-bed network #3 we needed the same packets, but with different volumes of traffic. Therefore, on the SmartBits packet generator we defined ten attack flows (from 1 Mbps up to 10Mbps, spaced at 1-Mbp intervals), using the two custom ICMP echo-request packets. This traffic gave us the ability to load the link J4300-switch on test-bed network #3 to its maximum capacity. Also, the 1-Mbp intervals produced more clear and useful results.

E. DATA COLLECTION

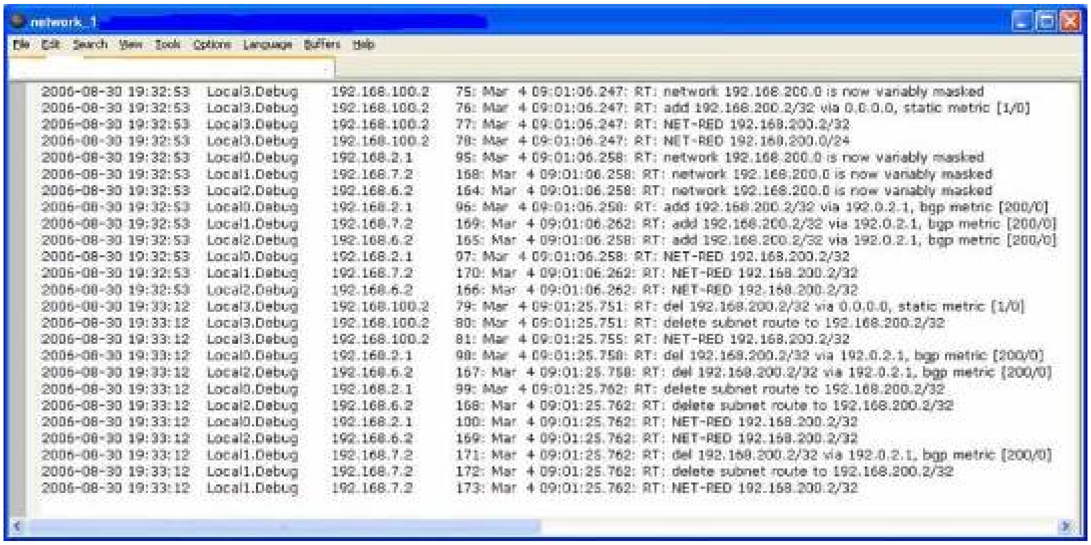
The main performance metric for this research was the response time of the routers. And the most accurate way to measure those values was to capture the trigger router’s initial routing-advertisement update and the border routers’ subsequent routing update messages. The messages were associated with the internal clock of each router. If the clocks were synchronized properly, the times for us to extract the information we needed would be accurate.

To synchronize the clocks in the routers, we used NTP protocol. The trigger router was configured to act as an NTP server; all the other routers were configured to take time information from the trigger router. This configuration gave us a millisecond precision that was very accurate for our measurements. Furthermore, since we didn’t evaluate the response time between each of the routers, but instead evaluated the response time as an average of all three of the border routers in comparison to different attack flows, any possible timing bias of a router would not affect the results.

The messages of the trigger router and the border routers were collected using the Simple Network Management Protocol (SNMP), the “debug” command on the Cisco routers, and the Kiwi Syslog Deamon version 8.0.2 application on the laptop PC. The “debug” command allows users to specify different kinds of messages to be produced by a router, to help administrators analyze, evaluate, or resolve a situation. The messages are created and transmitted according to the SNMP protocol. For our purposes, we used the “debug IP routing” and “debug BGP” options of the “debug” command that produce messages related to the routing information and to the BGP protocol.

The Kiwi Syslog Deamon is a freeware Windows application that receives, logs, and forwards Syslog messages from any Syslog-enabled device, like a router. Figure 18 presents an example of the messages captured during the application and the subsequent removal of destination-based Blackhole routing.

The IP addresses of the trigger router’s lone interface and the target host are 192.168.100.2 and 192.168.200.2, respectively. The addresses of the border routers’ trigger facing interfaces are 192.168.2.1, 192.168.7.2, and 192.168.6.2, respectively. In each border router the subnet 192.0.2.0/24 defines the static route to Null0. The address 192.0.2.1 is used by the trigger router as the “next-hop” in the BGP advertisements. The response time for a border router is the difference between the logging time of the trigger messages and the logging time of route update messages from that border router. In the example shown in Figure 18, the response time was a few milliseconds.



```
2006-08-30 19:32:53  Local3.Debug  192.168.100.2  75: Mar  4 09:01:06.247: RT: network 192.168.200.0 is now variably masked
2006-08-30 19:32:53  Local3.Debug  192.168.100.2  76: Mar  4 09:01:06.247: RT: add 192.168.200.2/32 via 0.0.0.0, static metric [1/0]
2006-08-30 19:32:53  Local3.Debug  192.168.100.2  77: Mar  4 09:01:06.247: RT: NET-RED 192.168.200.2/32
2006-08-30 19:32:53  Local3.Debug  192.168.100.2  78: Mar  4 09:01:06.247: RT: NET-RED 192.168.200.0/24
2006-08-30 19:32:53  Local0.Debug  192.168.2.1    95: Mar  4 09:01:06.258: RT: network 192.168.200.0 is now variably masked
2006-08-30 19:32:53  Local1.Debug  192.168.7.2    168: Mar  4 09:01:06.258: RT: network 192.168.200.0 is now variably masked
2006-08-30 19:32:53  Local2.Debug  192.168.6.2    164: Mar  4 09:01:06.258: RT: network 192.168.200.0 is now variably masked
2006-08-30 19:32:53  Local0.Debug  192.168.2.1    96: Mar  4 09:01:06.258: RT: add 192.168.200.2/32 via 192.0.2.1, bgp metric [200/0]
2006-08-30 19:32:53  Local1.Debug  192.168.7.2    169: Mar  4 09:01:06.262: RT: add 192.168.200.2/32 via 192.0.2.1, bgp metric [200/0]
2006-08-30 19:32:53  Local2.Debug  192.168.6.2    165: Mar  4 09:01:06.258: RT: add 192.168.200.2/32 via 192.0.2.1, bgp metric [200/0]
2006-08-30 19:32:53  Local0.Debug  192.168.2.1    97: Mar  4 09:01:06.258: RT: NET-RED 192.168.200.2/32
2006-08-30 19:32:53  Local1.Debug  192.168.7.2    170: Mar  4 09:01:06.262: RT: NET-RED 192.168.200.2/32
2006-08-30 19:32:53  Local2.Debug  192.168.6.2    166: Mar  4 09:01:06.262: RT: NET-RED 192.168.200.2/32
2006-08-30 19:33:12  Local3.Debug  192.168.100.2  79: Mar  4 09:01:25.751: RT: del 192.168.200.2/32 via 0.0.0.0, static metric [1/0]
2006-08-30 19:33:12  Local3.Debug  192.168.100.2  80: Mar  4 09:01:25.751: RT: delete subnet route to 192.168.200.2/32
2006-08-30 19:33:12  Local3.Debug  192.168.100.2  81: Mar  4 09:01:25.755: RT: NET-RED 192.168.200.2/32
2006-08-30 19:33:12  Local0.Debug  192.168.2.1    98: Mar  4 09:01:25.758: RT: del 192.168.200.2/32 via 192.0.2.1, bgp metric [200/0]
2006-08-30 19:33:12  Local2.Debug  192.168.6.2    167: Mar  4 09:01:25.758: RT: del 192.168.200.2/32 via 192.0.2.1, bgp metric [200/0]
2006-08-30 19:33:12  Local0.Debug  192.168.2.1    99: Mar  4 09:01:25.762: RT: delete subnet route to 192.168.200.2/32
2006-08-30 19:33:12  Local2.Debug  192.168.6.2    168: Mar  4 09:01:25.762: RT: delete subnet route to 192.168.200.2/32
2006-08-30 19:33:12  Local0.Debug  192.168.2.1    100: Mar  4 09:01:25.762: RT: NET-RED 192.168.200.2/32
2006-08-30 19:33:12  Local2.Debug  192.168.6.2    169: Mar  4 09:01:25.762: RT: NET-RED 192.168.200.2/32
2006-08-30 19:33:12  Local1.Debug  192.168.7.2    171: Mar  4 09:01:25.762: RT: del 192.168.200.2/32 via 192.0.2.1, bgp metric [200/0]
2006-08-30 19:33:12  Local1.Debug  192.168.7.2    172: Mar  4 09:01:25.762: RT: delete subnet route to 192.168.200.2/32
2006-08-30 19:33:12  Local1.Debug  192.168.7.2    173: Mar  4 09:01:25.762: RT: NET-RED 192.168.200.2/32
```

Figure 18. Example of debug messages after application and removal of destination-based Blackhole routing

The second performance metric we measured was the router CPU load. For this purpose we used the SolarWinds version 8.2 application, which is a network analysis, management, and monitoring software that comes with a variety of tools specified for different tasks. The most useful of these tools for monitoring the router CPU load is the “Router CPU load” version 8.0.15, which uses the SNMP protocol to query and get replies from properly configured routers. It then presents in graphical form the CPU load of every device either in real time or on average. Figure 19 shows an example of this tool during 7.09-Mbps total attack traffic in test-bed network #1.

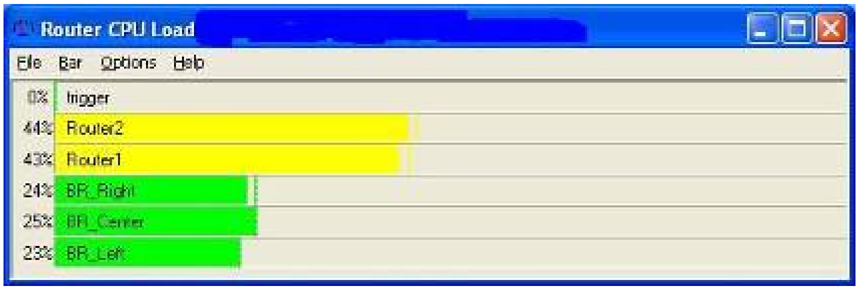
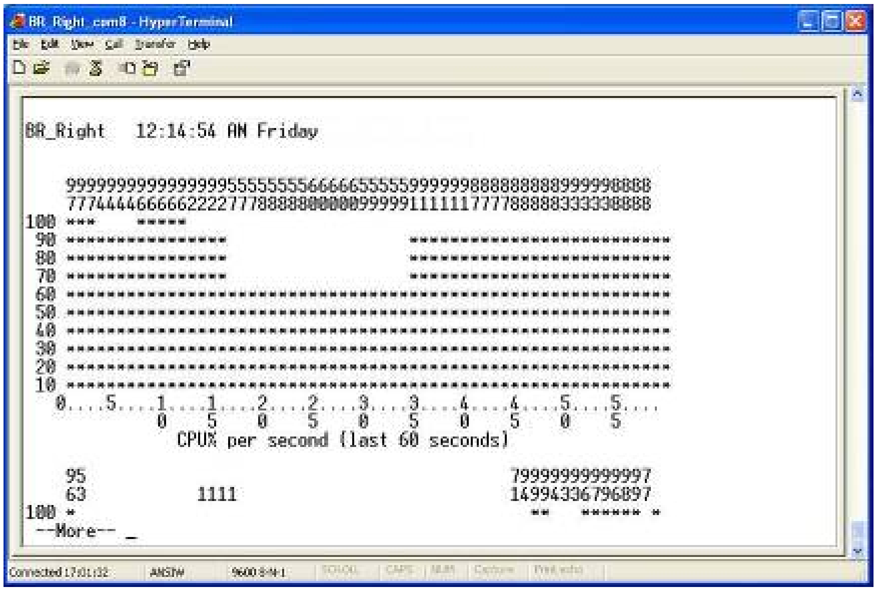


Figure 19. Example of 5 routers’ CPU loads under 7.09-Mbps total attack traffic

As noted above, for very high attack traffic (Flows #10–#13 in Table 1), the CPU load on some of the routers reached values of more than 90 percent. As a result, the “Router CPU load” tool couldn’t provide accurate measurements for these routers as SNMP depends on TCP to function and TCP breaks down when the CPU utilization of the routers is too high. For these cases we used a direct connection with the routers through Console port and Hyper Terminal. Then, by using the Cisco command “show processes cpu history", we were able to collect data about the CPU load for the last sixty seconds. Figure 20 presents an example of the printout.



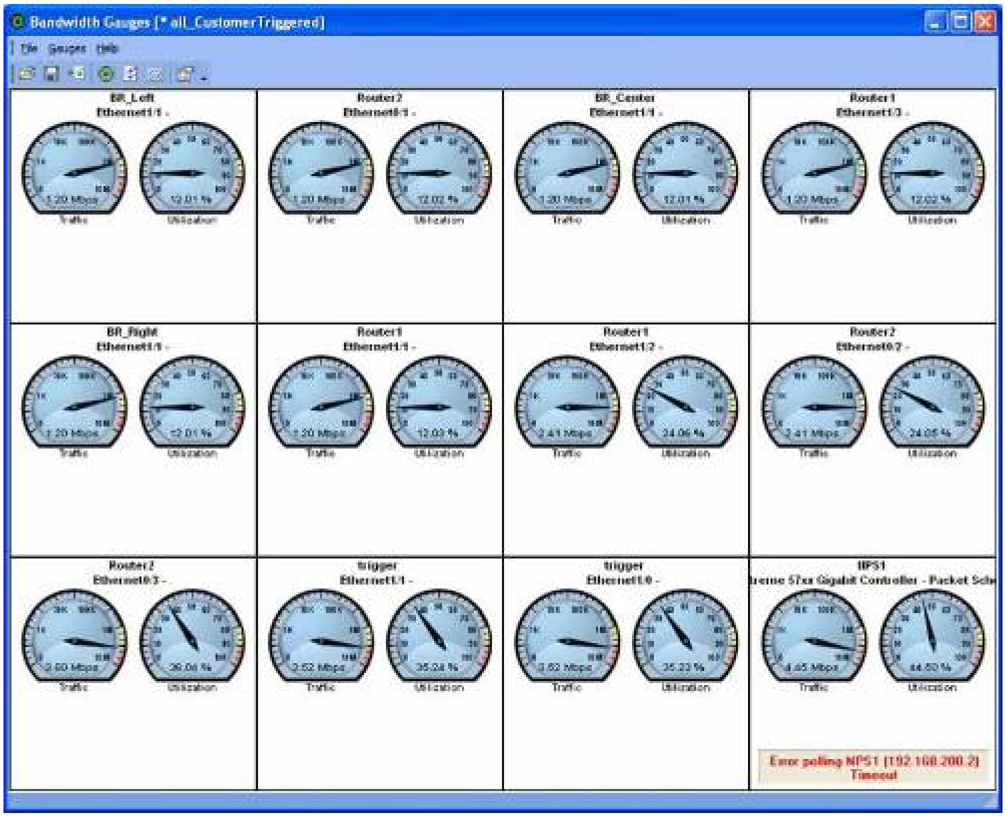
```
BR_Right   12:14:54 AM Friday


    999999999999999995555555566666555559999998888888888999998888
    777444466666222277788888800000999991111117777888888333338888
100 ***    *****
 90 *****************                ****************************
 80 *****************                ****************************
 70 *****************                ****************************
 60 ************************************************************
 50 ************************************************************
 40 ************************************************************
 30 ************************************************************
 20 ************************************************************
 10 ************************************************************
   0....5....1....1....2....2....3....3....4....4....5....5....
             0    5    0    5    0    5    0    5    0    5
               CPU% per second (last 60 seconds)

    95                                           79999999999997
    63          1111                             14994336796897
100 *                                              **   ****** *
 --More-- _
```

Figure 20. CPU load history on Cisco routers

From the SolarWinds application, we also used the “Bandwidth Gauges” version 8.0.26 tool, which, by using the SNMP protocol again, presented real-time traffic-load monitoring. For each device we specified the interface we were interested in and the tool showed the average traffic (logarithmic or linear bps) and the average percent utilization. Figure 21 presents an example of the printout during 7.09-Mbps attack traffic.



Error polling IPS1 (192.168.200.2)
Timeout

Figure 21. Example of Bandwidth Gauges tool during 7.09-Mbps attack traffic

In this chapter we explained the methodology we used in our research. We described the hardware and software we used to create the DDoS attack traffic and the applications we used to collect the data. In the next chapter we present the final results of this research.

IV. RESULTS

A. INTRODUCTION

In this chapter we present the results of our research. In the first section, we present an evaluation of the three basic BGP Blackhole routing methods (RTBH destination-based, RTBH source-based, and customer-triggered Blackhole routing) under low to medium attack traffic as compared to the capabilities of the routers inside our test-bed network.

In the second section, we present an evaluation of the same methods under extreme traffic. This division of the evaluations into two sections is necessary for two reasons. First, the unstable behavior of the routers imposed different approaches for collecting the data required for the evaluations. Second, the results in the evaluation of the high-traffic cases were in many ways different from those under low-to-medium attack traffic.

The third section of this evaluation concerns the performance of BGP Blackhole routing in a network where the limiting factor is not the routers’ CPU load, as in the previous two cases, but instead is the link load.

In the final section of this chapter, we compare the preconfiguration of the routers for the three basic BGP Blackhole routing methods. To achieve this goal, we examine the effect of the preconfiguration on the routers before the application of any BGP Blackhole routing.

B. COMPARISON OF BASIC BGP BLACKHOLE ROUTING METHODS UNDER LOW-TO-MEDIUM ATTACK TRAFFIC

From test-bed networks #1 and #2, the data collected was related to the response time of the three border routers under varying amounts of attack traffic. As noted in the previous chapter, we specified thirteen different flows in the packet generator. The first nine flows were simulating low-to-medium attack traffic for the specific routers we used in the test-bed networks. The three basic BGP Blackhole routing methods were tested against those nine flows. In order to achieve more accurate results, each experiment was run fifteen times, and then the average of the response time was calculated for every border router. Figure 22 shows the performance of these methods under the first nine attack flows.

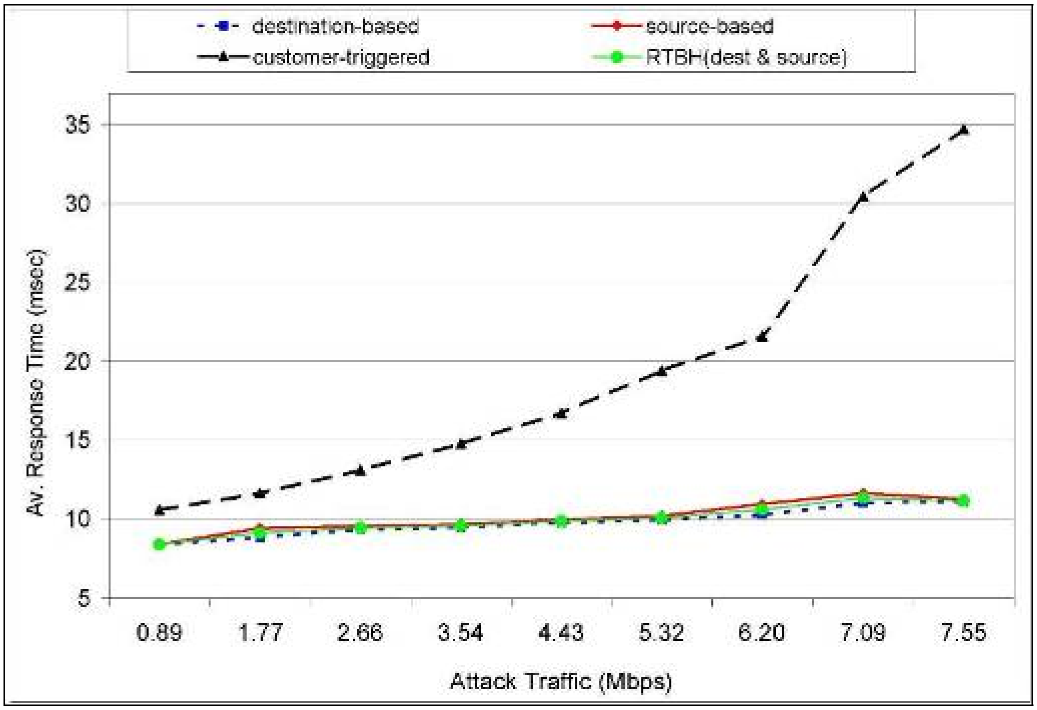


Figure 22. Routers’ response time in the three basic BGP Blackhole routing methods under low-to--medium attack traffic

This diagram shows clearly the degraded performance of the customer-triggered method as compared to the remote-triggered methods of BGP Blackhole routing. This difference can be explained by the fact that the trigger’s iBGP advertisements in the case of customer-triggered BGP Blackhole routing have to pass through the same loaded link that the attack traffic uses to approach the target.

The destination-based and source-based methods had almost identical performances which remained relatively constant for all nine flows. Figure 23 shows the percentage of increase in response time with customer-triggered Blackhole routing, relative to the average response time of RTBH (destination- and source-based) routing. The maximum value is more than 200 percent of the response time in RTBH routing. Another important observation is that the response time of the RTBH routing in the highest flow is only slightly larger than the response time of the customer-triggered Blackhole routing in the lowest flow (11.6 ms versus 10.5ms).

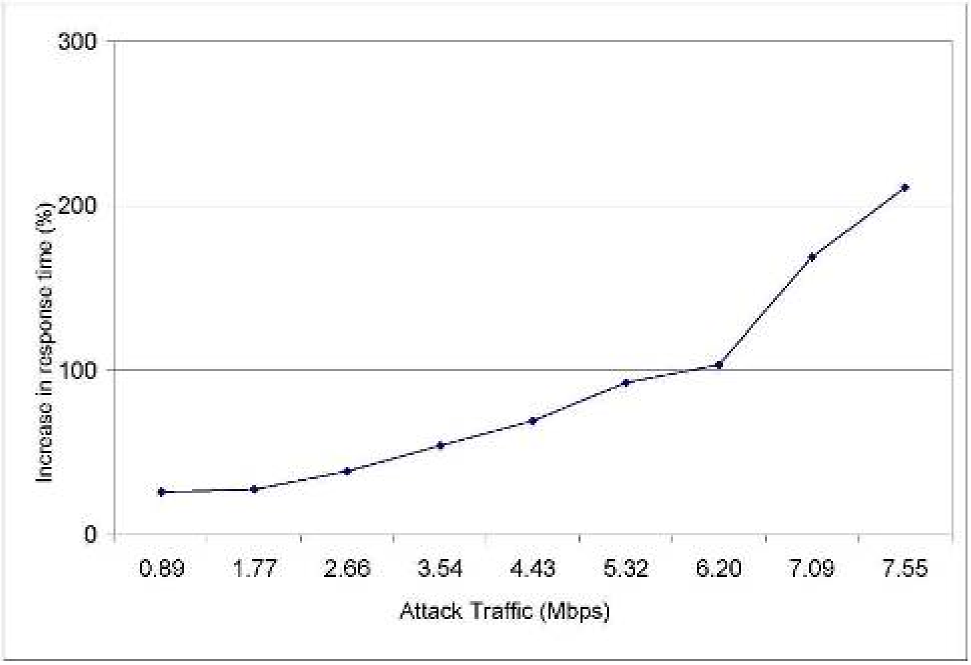
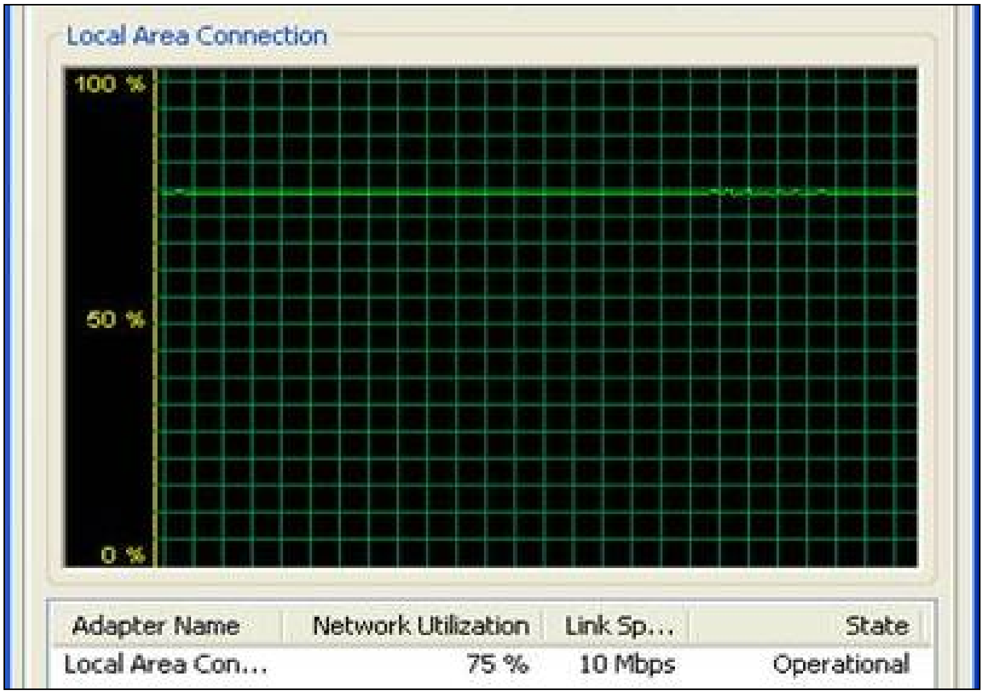


Figure 23. Increase in response time (%) with customer-triggered BH routing

C. COMPARISON OF BGP BLACKHOLE ROUTING METHODS UNDER DIFFERENT ROUTER CPU LOADS

As noted previously, in test-bed networks #1 and #2 when the attack traffic exceeded 7.55-Mbps volume, the behavior of the routers became unstable and the methodology of data collection had to be different.

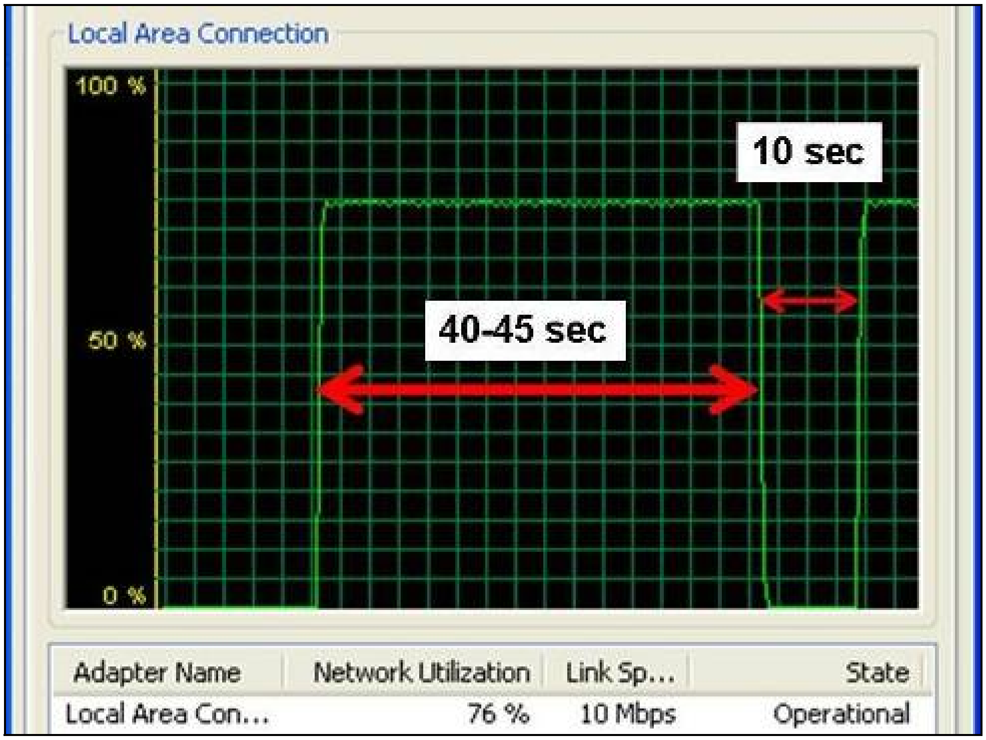
The basic difference between the low-to-medium and the high attack traffic was observable mainly on the target. In the former, all the attack traffic could approach the target at a constant rate. Figure 24 shows the Microsoft Windows Task Manager-Network tab on the target PC under 7.55-Mbps attack traffic.



Adapter Name	Network Utilization	Link Sp...	State
Local Area Con...	75 %	10 Mbps	Operational

Figure 24. Network utilization on target PC under 7.55-Mbps attack traffic

In the case of high attack traffic, the utilization of the network interface on the target host followed a periodical pattern consisting of two parts. The first part had a duration of 40-45 seconds. During this period the utilization was 75-76 percent, continuously. This means that only 7.5-Mbps to 7.6-Mbps traffic was approaching the target no matter how high the attack traffic that was being created by the packet generator. During the second part, which lasted for 10 seconds, the utilization was zero and no traffic was able to reach the target. The most probable reason for this behavior is queuing and/or CPU constraints in the routers. This pattern is presented in Figure 25.



10 sec

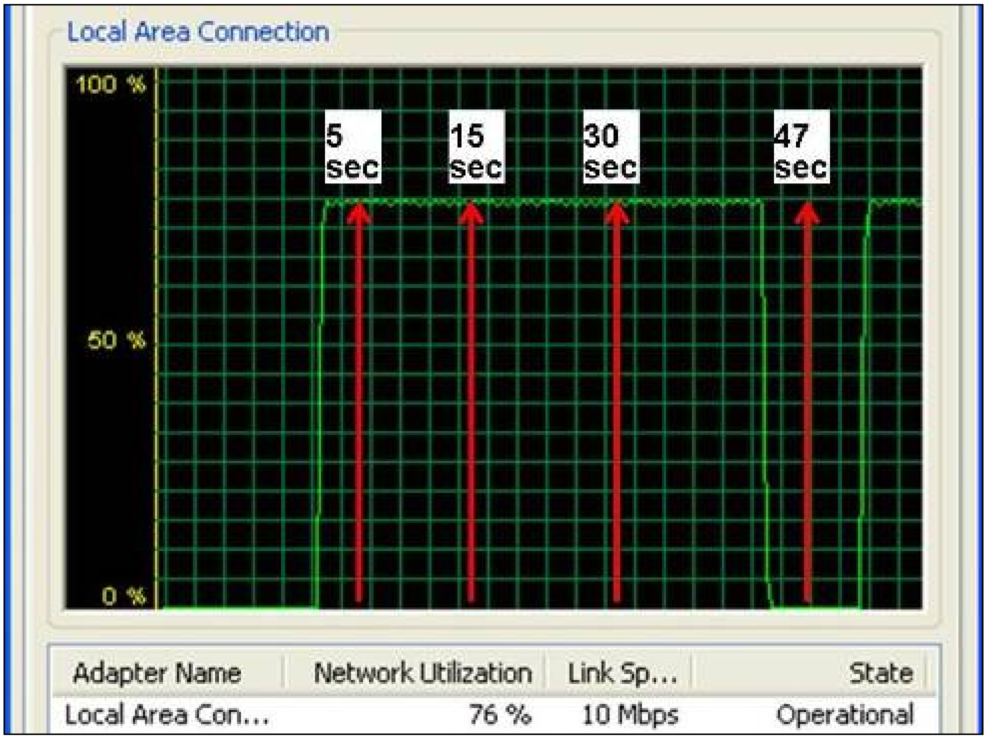
40-45 sec

Adapter Name	Network Utilization	Link Sp...	State
Local Area Con...	76 %	10 Mbps	Operational

Figure 25. Network utilization on target PC under high attack traffic

The pattern was repeated continuously for as long as the attack traffic was being created by the packet generator or until the Blackhole routing updates from the trigger successfully reached the border routers.

The application of Blackhole routing had varying results on response time depending on how many seconds after the initialization of attack the appropriate commands were issued on the trigger router. For this reason we applied Blackhole routing at four time instances — 5, 15, 30, and 47 seconds — within one occurrence of the load pattern, and we measured the response time for those four cases. Figure 26 shows where the four time instances are relative to the load pattern.



Adapter Name	Network Utilization	Link Sp...	State
Local Area Con...	76 %	10 Mbps	Operational

Figure 26. Time instances of applying Blackhole routing at the trigger

For traffic flows #10 to #13 from Table 1, we ran four simulations, one for each time instance. Every simulation was repeated ten times. The average response times corresponding to the different time instances are presented in Figures 27, 28, 29, and 30.

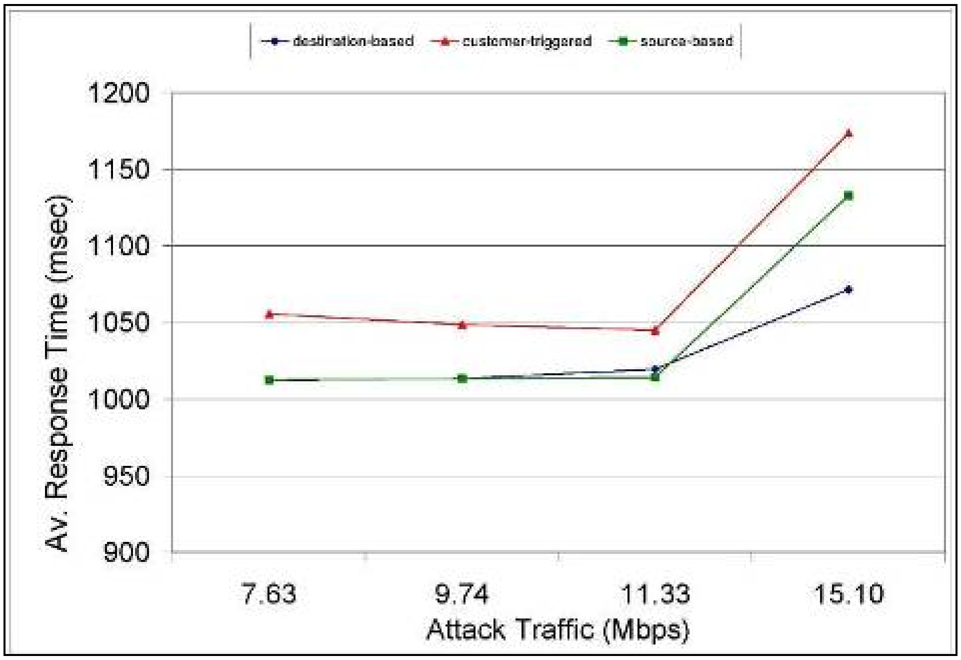


Figure 27. Average response times in 5-second time instances under high attack traffic

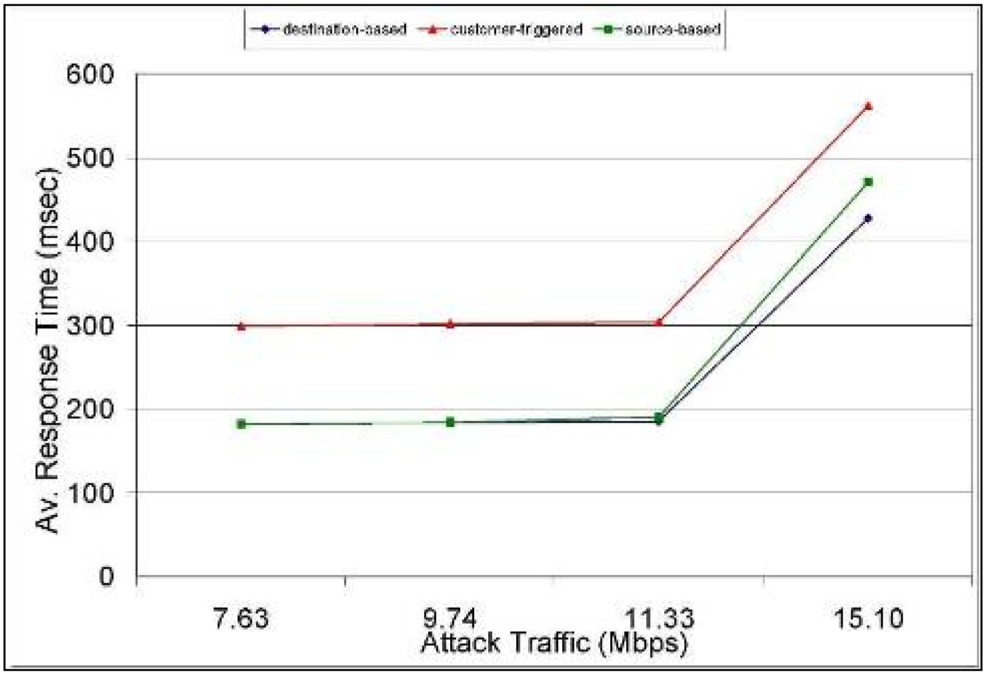


Figure 28. Average response times in 15-second time instances under high attack traffic

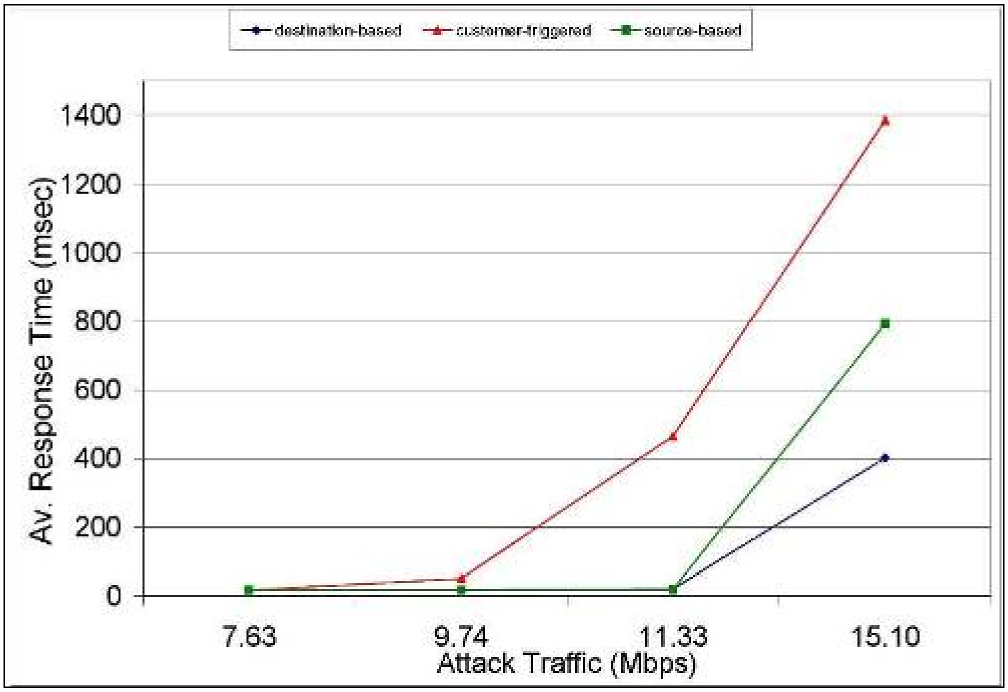


Figure 29. Average response times in 30-second time instances under high attack traffic

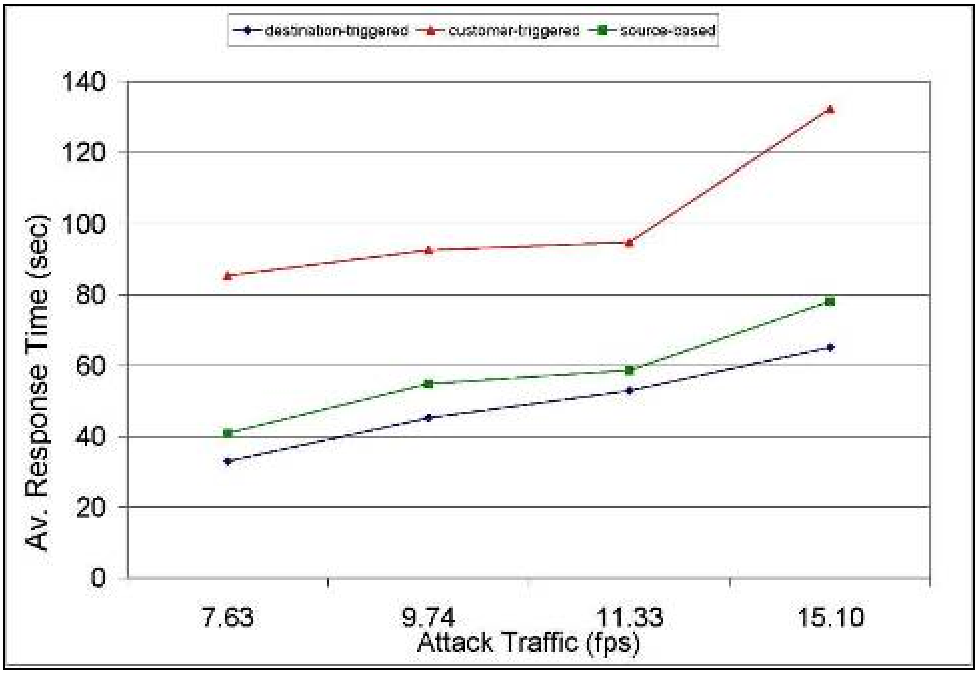


Figure 30. Average response times in 45-second time instances under high attack traffic

From the above diagrams it is obvious that the worst case scenario for applying Blackhole routing was the 10-second window of zero utilization seen in Figure 25. The average response time for this case was in seconds, while in the other three cases the response time was in milliseconds. During our measurements the maximum value of response time that was recorded was 295 seconds (≈4.9 minutes), which occurred in the customer-triggered Blackhole routing.

Figure 31 shows the average response time from all methods related to the time slot that Blackhole routing was initiated. The diagram shows that for every periodic pattern, initially the response time dropped from about 1 second to about 250 milliseconds. Then there is the 10-second period in which the response time exceeds values of 1 minute.

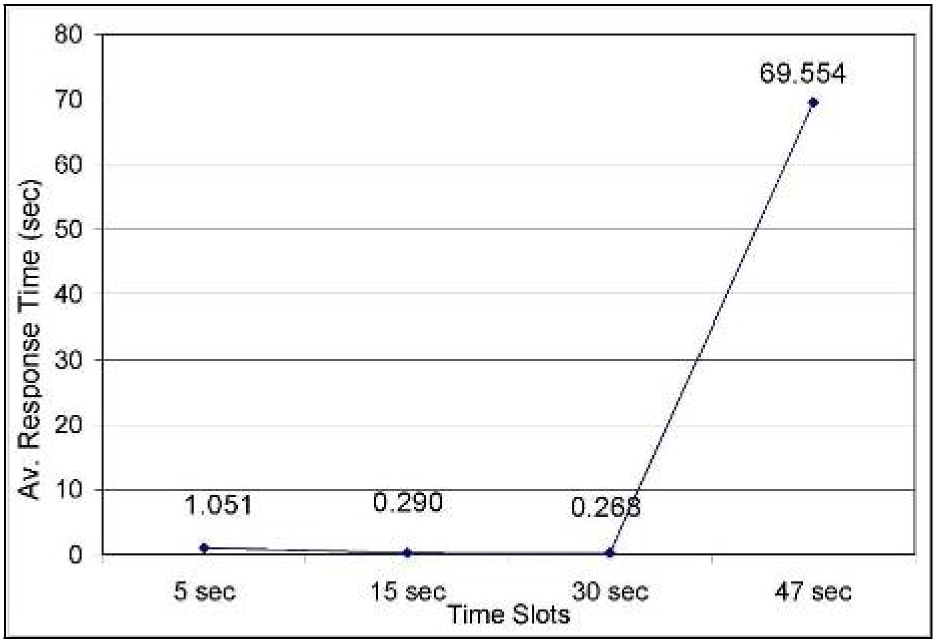


Figure 31. Average response time for every time slot

A summary of Figures 27-30 is presented in Figure 32.

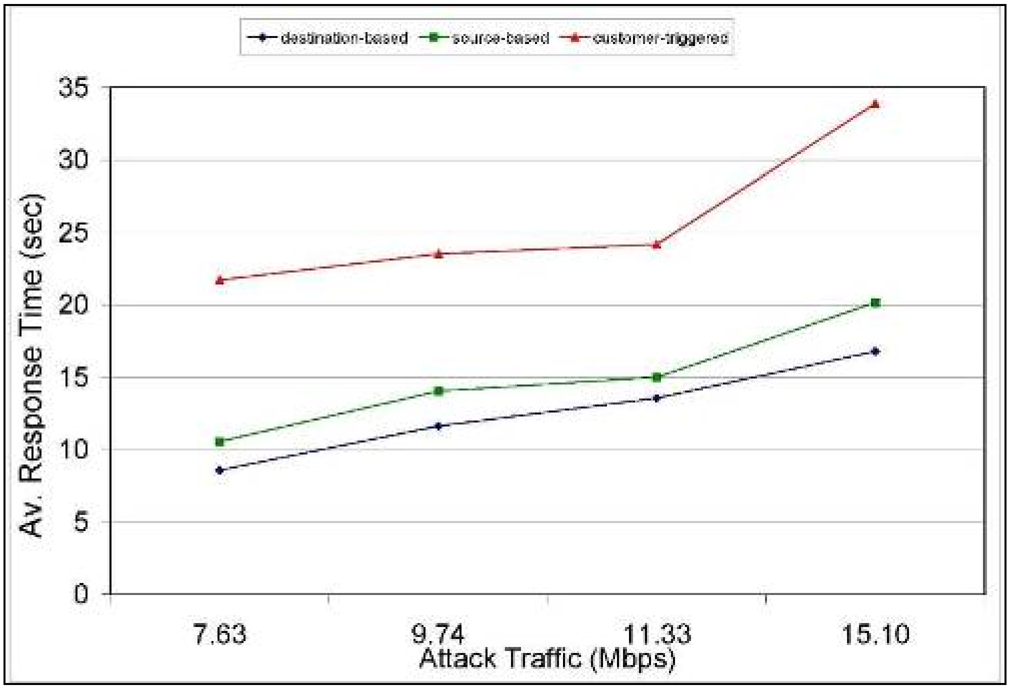


Figure 32. Average response time in high attack traffic

As with the low-to-medium attack traffic, the customer-triggered Blackhole method presents the worst performance. Figure 33 shows the average increase in response time with customer-triggered Blackhole routing as a percentage of the average response time of RTBH routing (destination- and source-based). The total average increase in response time is 90.92 percent. This means that when the customer-triggered method is used under high attack traffic, the response time will, on average, be 90.92 percent larger, as compared to the response time when the other two methods are used.

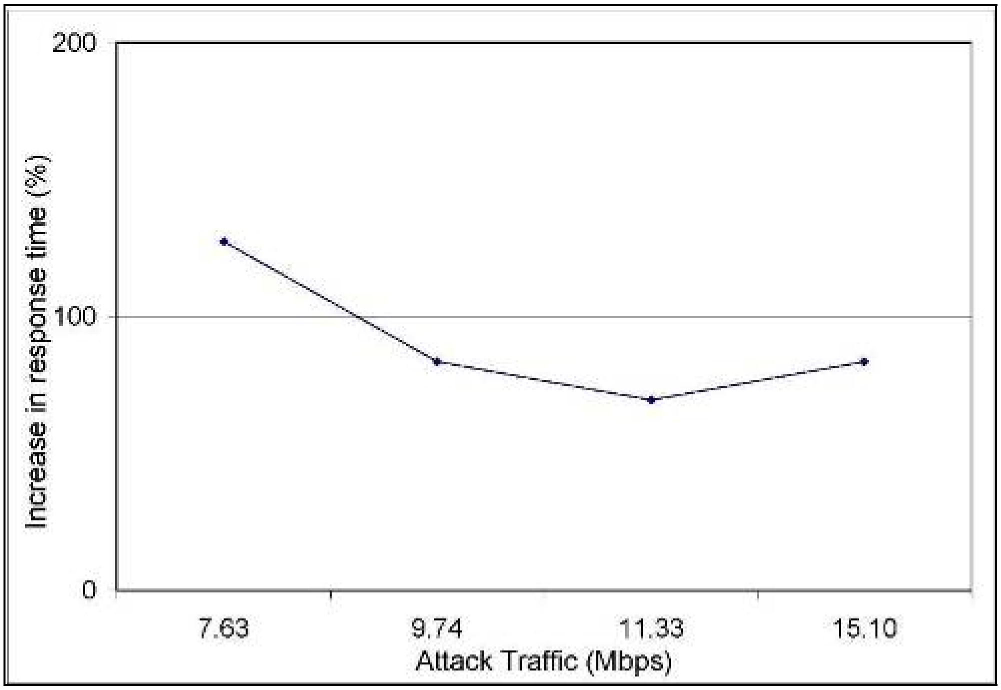


Figure 33. Percentage of increased response time with customer-triggered BH routing as compared to the average of the RTBH routing methods

Figure 32 shows that, compared to the destination-based Blackhole routing, source-based Blackhole routing also had degraded performance. In the low-to-medium attack traffic this was not the case: the performance for both methods was almost the same. Figure 34 shows the percentage of increase in response time with source-based Blackhole routing, as compared to destination-based Blackhole routing. The average total increase was 18.68 percent.

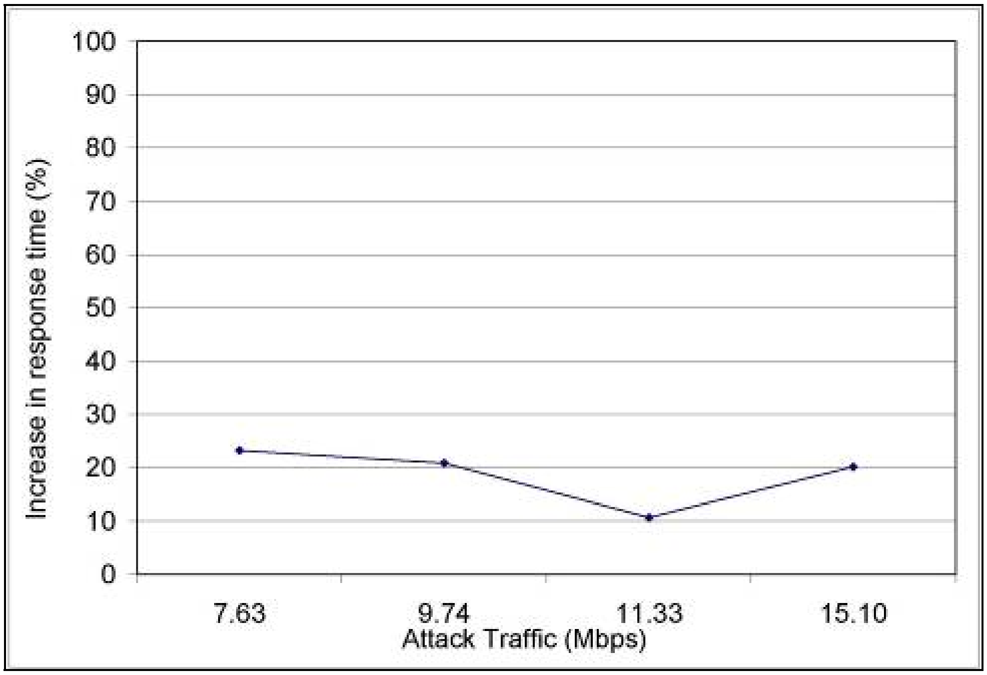


Figure 34. Percentage of increased response time with source-based BH routing as compared to the destination-based BH routing

As noted previously, the collection of data related to the routers' CPU load wasn't possible with the methods we used for the low-to-medium attack traffic. The highly unstable performance of the routers in high attack traffic indicates that the CPU load in some of the routers inside the network reached very high values. For this reason, we used the methodology explained in the previous chapter to calculate the CPU load of the border routers and the router closest to the target, which was obviously the bottleneck in both test-bed networks #1 and #2.

Figure 35 presents a sample of the CPU load of those routers under high attack traffic. The diagram shows that for every periodic pattern after the first 30 seconds the CPU load in the bottleneck router exceeds 90 percent, and this fact explains the degraded performance of the Blackhole routing in this time area.

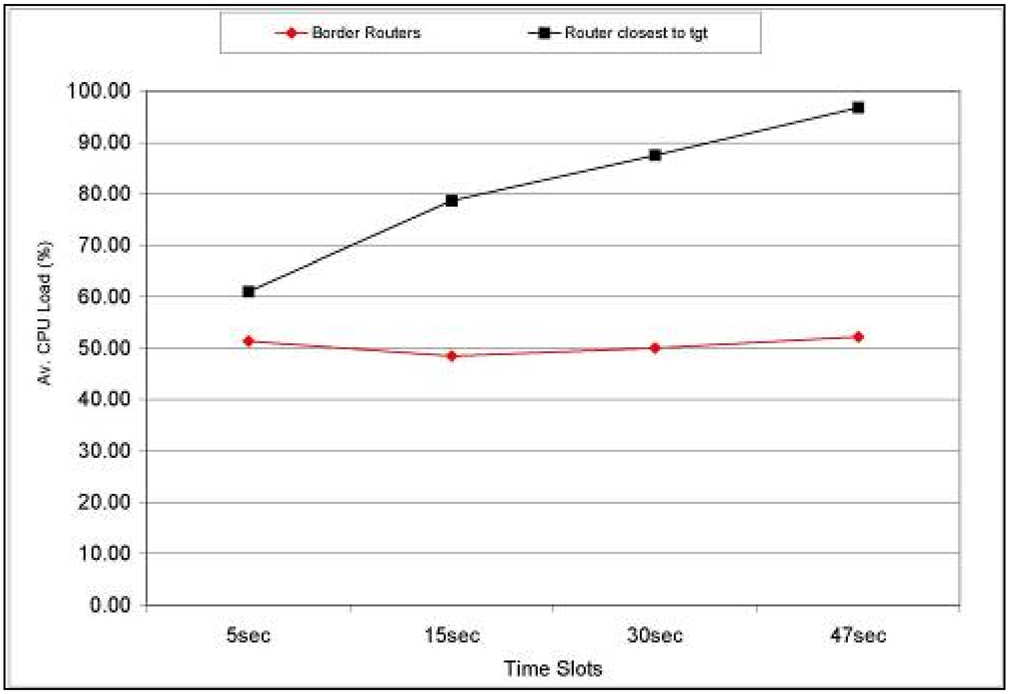


Figure 35. Sample of routers’ CPU load under high attack traffic

In summary, the results show that a very high CPU load on a router in the path used by the iBGP session can significantly delay Blackhole routing. Furthermore, the data clearly shows that the employment of the Blackhole routing must be as soon as possible after the initiation of the attack. The destination-based RTBH routing method performs the best in such situations.

D. EVALUATION OF BGP BLACKHOLE ROUTING UNDER A HIGH LINK LOAD

For the evaluation of BGP Blackhole routing under a high link load we used test-bed #3 and ten attack flows, from 1-Mbps up to 10-Mbps. During the initial measurements we realized that there was a timing boundary that divided the performance of Blackhole routing into two cases. The boundary occurred approximately 40 seconds after the initialization of the attack.

Before the timing boundary Blackhole routing had a relatively constant response time of about 1515 milliseconds. Figure 36 shows the average response time from our measurements.

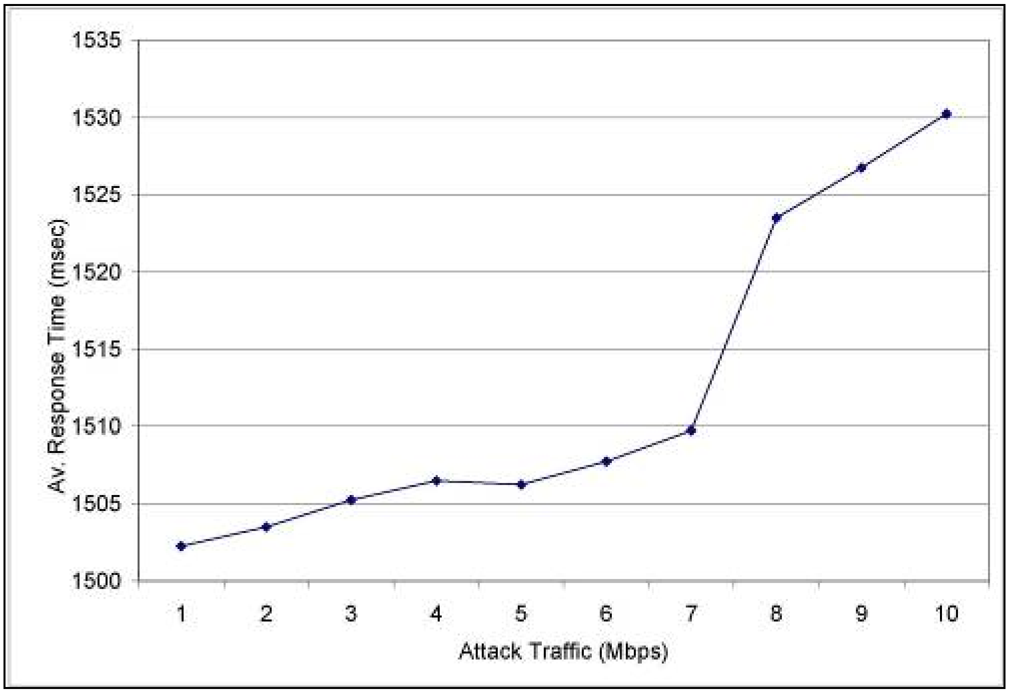


Figure 36. Response time before the 40-second boundary

After the 40-second boundary the performance of the Blackhole routing was highly degraded. Figure 37 shows the performance of Blackhole routing in this case.

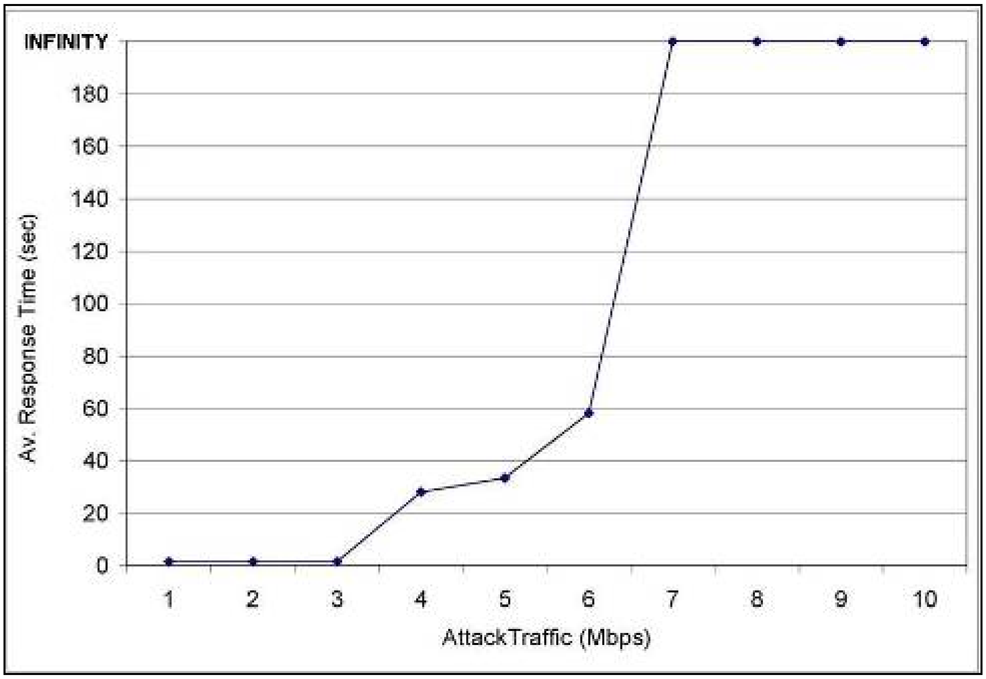
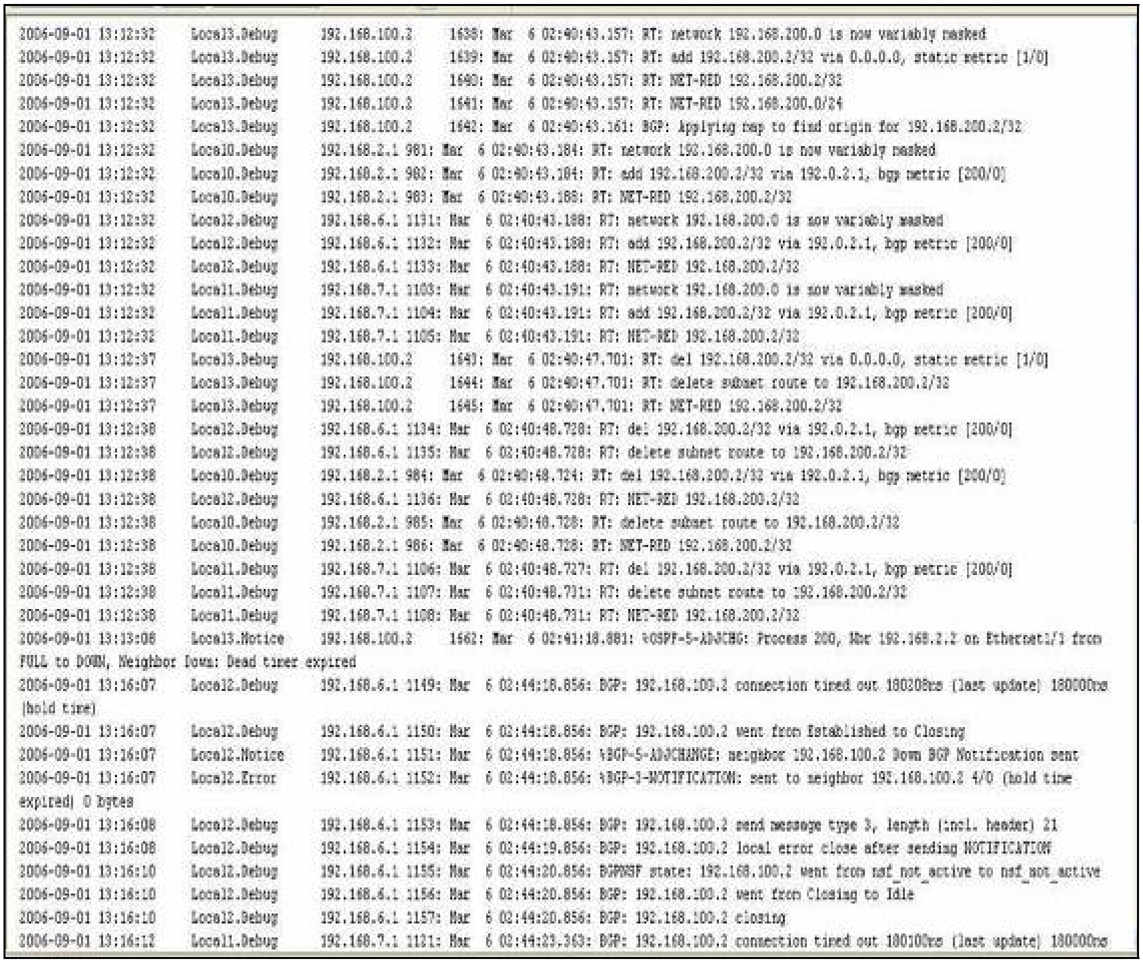


Figure 37. Response time after the 40-second boundary

For 1-Mbps to 3-Mbps attack traffic the response time had the same values as in the first case, approximately 1.5 seconds. For 4-Mbps to 6-Mbps attack traffic the response time varied from 30 seconds to nearly 1 minute. For attack traffic of 7-Mbps or higher the response time was infinity.

To confirm the latter result we run multiple simulations for 20-minute period. The Blackhole advertisements were not able to reach the border routers in any of them. Figure 38 shows the Syslog routers' messages in one of those simulations.



```
2006-09-01 13:12:32    Local3.Debug    192.168.100.2     1638: Mar  6 02:40:43.157: RT: network 192.168.200.0 is now variably masked
2006-09-01 13:12:32    Local3.Debug    192.168.100.2     1639: Mar  6 02:40:43.157: RT: add 192.168.200.2/32 via 0.0.0.0, static metric [1/0]
2006-09-01 13:12:32    Local3.Debug    192.168.100.2     1640: Mar  6 02:40:43.157: RT: NET-RED 192.168.200.2/32
2006-09-01 13:12:32    Local3.Debug    192.168.100.2     1641: Mar  6 02:40:43.157: RT: NET-RED 192.168.200.0/24
2006-09-01 13:12:32    Local3.Debug    192.168.100.2     1642: Mar  6 02:40:43.161: BGP: Applying map to find origin for 192.168.200.2/32
2006-09-01 13:12:32    Local0.Debug    192.168.2.1 981: Mar  6 02:40:43.184: RT: network 192.168.200.0 is now variably masked
2006-09-01 13:12:32    Local0.Debug    192.168.2.1 982: Mar  6 02:40:43.184: RT: add 192.168.200.2/32 via 192.0.2.1, bgp metric [200/0]
2006-09-01 13:12:32    Local0.Debug    192.168.2.1 983: Mar  6 02:40:43.188: RT: NET-RED 192.168.200.2/32
2006-09-01 13:12:32    Local2.Debug    192.168.6.1 1131: Mar  6 02:40:43.188: RT: network 192.168.200.0 is now variably masked
2006-09-01 13:12:32    Local2.Debug    192.168.6.1 1132: Mar  6 02:40:43.188: RT: add 192.168.200.2/32 via 192.0.2.1, bgp metric [200/0]
2006-09-01 13:12:32    Local2.Debug    192.168.6.1 1133: Mar  6 02:40:43.188: RT: NET-RED 192.168.200.2/32
2006-09-01 13:12:32    Local1.Debug    192.168.7.1 1103: Mar  6 02:40:43.191: RT: network 192.168.200.0 is now variably masked
2006-09-01 13:12:32    Local1.Debug    192.168.7.1 1104: Mar  6 02:40:43.191: RT: add 192.168.200.2/32 via 192.0.2.1, bgp metric [200/0]
2006-09-01 13:12:32    Local1.Debug    192.168.7.1 1105: Mar  6 02:40:43.191: RT: NET-RED 192.168.200.2/32
2006-09-01 13:12:37    Local3.Debug    192.168.100.2     1643: Mar  6 02:40:47.701: RT: del 192.168.200.2/32 via 0.0.0.0, static metric [1/0]
2006-09-01 13:12:37    Local3.Debug    192.168.100.2     1644: Mar  6 02:40:47.701: RT: delete subnet route to 192.168.200.2/32
2006-09-01 13:12:37    Local3.Debug    192.168.100.2     1645: Mar  6 02:40:47.701: RT: NET-RED 192.168.200.2/32
2006-09-01 13:12:38    Local2.Debug    192.168.6.1 1134: Mar  6 02:40:48.728: RT: del 192.168.200.2/32 via 192.0.2.1, bgp metric [200/0]
2006-09-01 13:12:38    Local2.Debug    192.168.6.1 1135: Mar  6 02:40:48.728: RT: delete subnet route to 192.168.200.2/32
2006-09-01 13:12:38    Local0.Debug    192.168.2.1 984: Mar  6 02:40:48.724: RT: del 192.168.200.2/32 via 192.0.2.1, bgp metric [200/0]
2006-09-01 13:12:38    Local2.Debug    192.168.6.1 1136: Mar  6 02:40:48.728: RT: NET-RED 192.168.200.2/32
2006-09-01 13:12:38    Local0.Debug    192.168.2.1 985: Mar  6 02:40:48.728: RT: delete subnet route to 192.168.200.2/32
2006-09-01 13:12:38    Local0.Debug    192.168.2.1 986: Mar  6 02:40:48.728: RT: NET-RED 192.168.200.2/32
2006-09-01 13:12:38    Local1.Debug    192.168.7.1 1106: Mar  6 02:40:48.727: RT: del 192.168.200.2/32 via 192.0.2.1, bgp metric [200/0]
2006-09-01 13:12:38    Local1.Debug    192.168.7.1 1107: Mar  6 02:40:48.731: RT: delete subnet route to 192.168.200.2/32
2006-09-01 13:12:38    Local1.Debug    192.168.7.1 1108: Mar  6 02:40:48.731: RT: NET-RED 192.168.200.2/32
2006-09-01 13:13:08    Local3.Notice   192.168.100.2     1662: Mar  6 02:41:18.881: %OSPF-5-ADJCHG: Process 200, Nbr 192.168.2.2 on Ethernet1/1 from
FULL to DOWN, Neighbor Down: Dead timer expired
2006-09-01 13:16:07    Local2.Debug    192.168.6.1 1149: Mar  6 02:44:18.856: BGP: 192.168.100.2 connection timed out 180208ms (last update) 180000ms
(hold time)
2006-09-01 13:16:07    Local2.Debug    192.168.6.1 1150: Mar  6 02:44:18.856: BGP: 192.168.100.2 went from Established to Closing
2006-09-01 13:16:07    Local2.Notice   192.168.6.1 1151: Mar  6 02:44:18.856: %BGP-5-ADJCHANGE: neighbor 192.168.100.2 Down BGP Notification sent
2006-09-01 13:16:07    Local2.Error    192.168.6.1 1152: Mar  6 02:44:18.856: %BGP-3-NOTIFICATION: sent to neighbor 192.168.100.2 4/0 (hold time
expired) 0 bytes
2006-09-01 13:16:08    Local2.Debug    192.168.6.1 1153: Mar  6 02:44:18.856: BGP: 192.168.100.2 send message type 3, length (incl. header) 21
2006-09-01 13:16:08    Local2.Debug    192.168.6.1 1154: Mar  6 02:44:19.856: BGP: 192.168.100.2 local error close after sending NOTIFICATION
2006-09-01 13:16:10    Local2.Debug    192.168.6.1 1155: Mar  6 02:44:20.856: BGPNSF state: 192.168.100.2 went from nsf_not_active to nsf_not_active
2006-09-01 13:16:10    Local2.Debug    192.168.6.1 1156: Mar  6 02:44:20.856: BGP: 192.168.100.2 went from Closing to Idle
2006-09-01 13:16:10    Local2.Debug    192.168.6.1 1157: Mar  6 02:44:20.856: BGP: 192.168.100.2 closing
2006-09-01 13:16:12    Local1.Debug    192.168.7.1 1121: Mar  6 02:44:23.363: BGP: 192.168.100.2 connection timed out 180100ms (last update) 180000ms
```

Figure 38. Syslog messages during a high-link-load BH routing application

In this example the attack traffic was initiated at 02:40:40. After 3 seconds we applied Blackhole routing that was successfully advertised and the response time was a few milliseconds. Seven seconds after the attack we successfully removed the Blackhole routing and the response time was slightly above 1 second. At 02:41:18, 38 seconds after the attack, the trigger router sent an OSPF message saying the neighbor was down, meaning that the connection to the network was down. At 02:44:18, 120 seconds after the last OSPF message, the other routers started sending BGP messages declaring the various connections to their BGP neighbors, from Established to Closing.

This sequence explains the problematic performance of BGP Blackhole routing under a very high link load. The default “dead-timer” for OSPF is 40 seconds. If there is no reply during this period, OSPF considers the links dead and updates the routing table by removing the related entries. That causes the TCP to fail and, since the iBGP session uses TCP, it also fails. The default “keep-alive” timer for BGP is 120 seconds. After that period, BGP declares that the connections to neighbors are closed. The final result is that there are no BGP updates for as long as the situation remains the same, which means that there is no way to successfully apply BGP Blackhole routing.

This was one of the most important results of our research because it showed for the first time that there are situations in which BGP Blackhole routing will not be at all effective.

E. EVALUATION OF THE ROUTERS’ PRECONFIGURATION FOR THE THREE BASIC BGP BLACKHOLE ROUTING METHODS

To evaluate the effect of Blackhole routing pre-configuration on a router, we applied attack traffic in test-bed networks #1 and #2. Then we measured the CPU load for every router before the initialization of the Blackhole routing. Figures 39, 40, and 41 present these measurements.

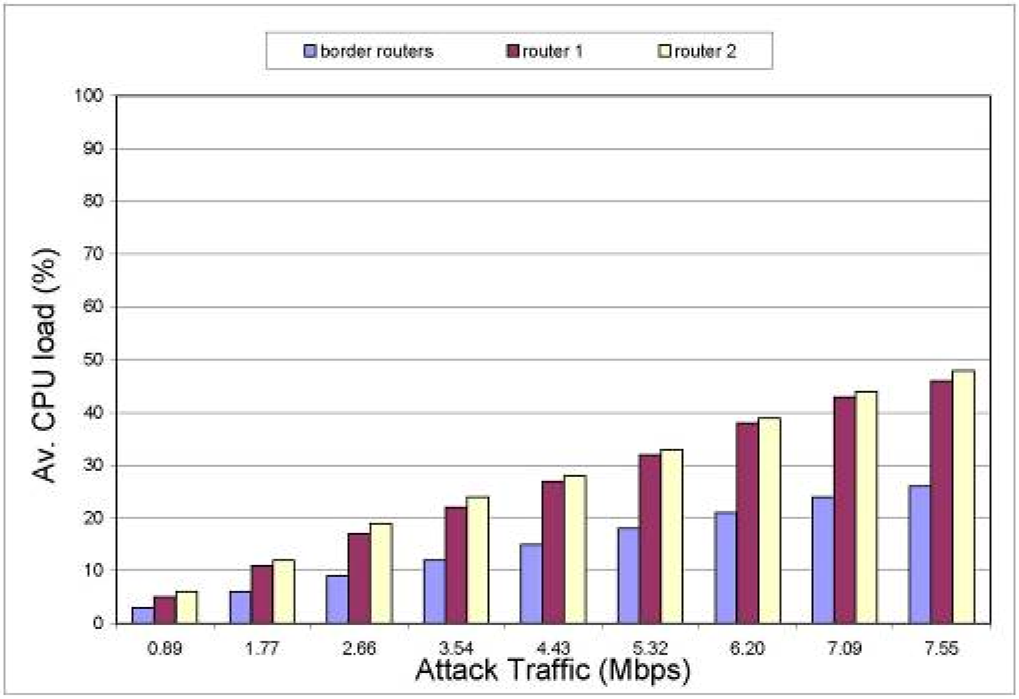


Figure 39. Routers' CPU load before application of destination-based RTBH routing

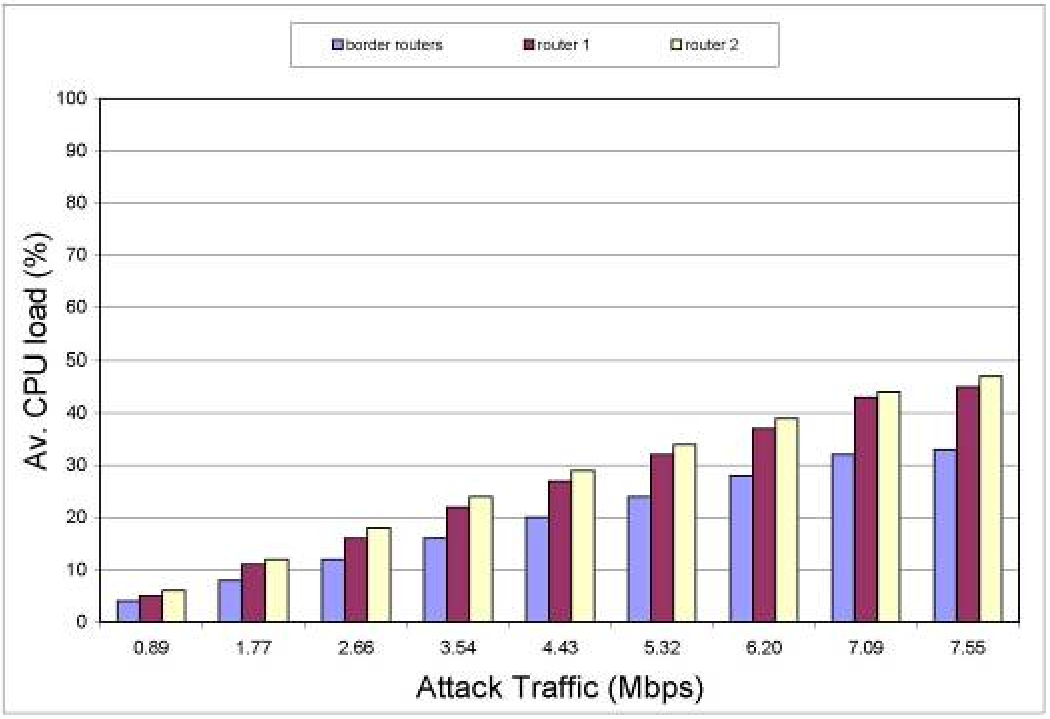


Figure 40. Routers' CPU load before application of source-based RTBH routing

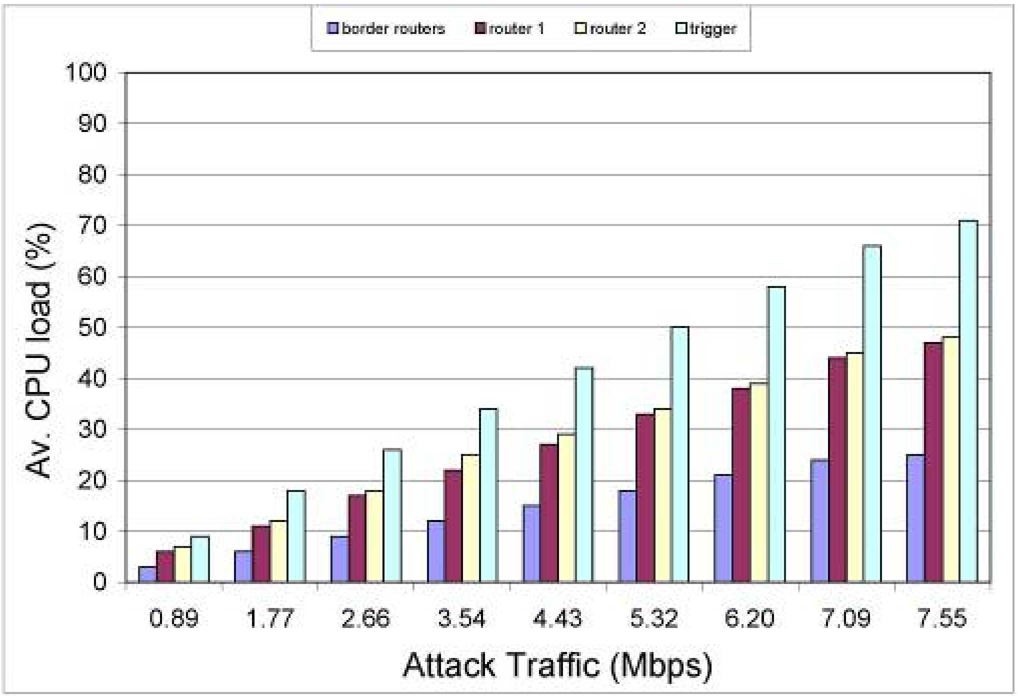


Figure 41. Routers’ CPU load before application of customer-triggered Blackhole routing

In the above three diagrams, Router 2, though it handles 50 percent more traffic than Router 1, presents only a slight increase in its CPU load as compared to Router 1. In Figure 41 the trigger router presents a significantly greater CPU load as compared to Router 2, though both handle the same amount of traffic. This difference can be explained by the fact that Router 2 was a more capable model (Cisco 3600) than the other routers (Cisco 2621XMs). The difference in the CPU load of these two models is presented in Figures 42 and 43.

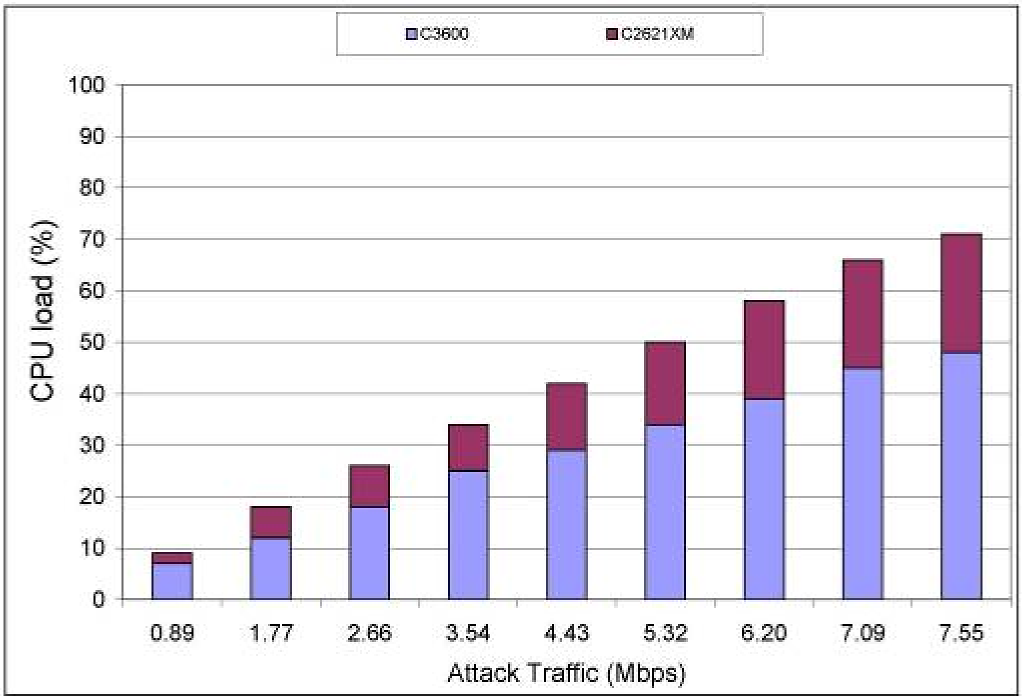


Figure 42. Difference in CPU load of the Cisco 2621XM router and the Cisco 3600 router under the same traffic

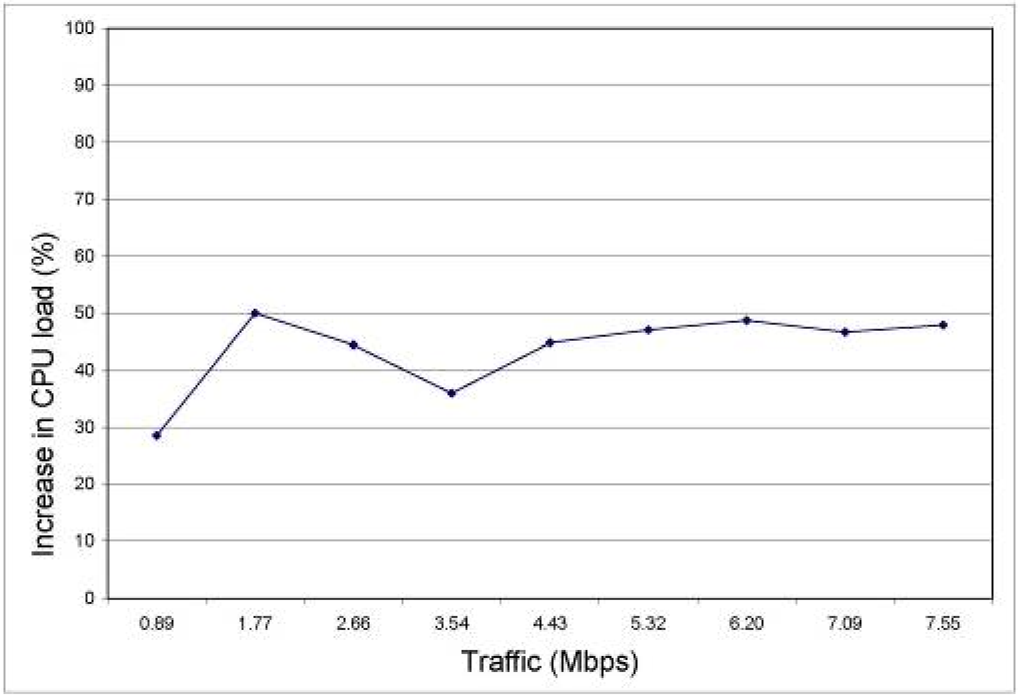


Figure 43. Percentage of increased CPU load in the Cisco 2621XM router as compared to the Cisco 3600 router

For a maximum attack traffic of 7.55-Mbps, the difference in the CPU load reached 23 percent. On average, the Cisco 2621XM router had a 44-percent larger CPU load than the Cisco 3600 router under the same traffic.

As noted previously, the behavior of the routers under more than 7.55-Mbps attack traffic became unstable and the response time increased significantly. This can be explained partly as a result of the lower performance of the Cisco 2621XM model. It also shows how important the performance of the routers used in a network is in defending against DDoS attacks.

Figures 44, 45, and 46 summarize the CPU load for every router.

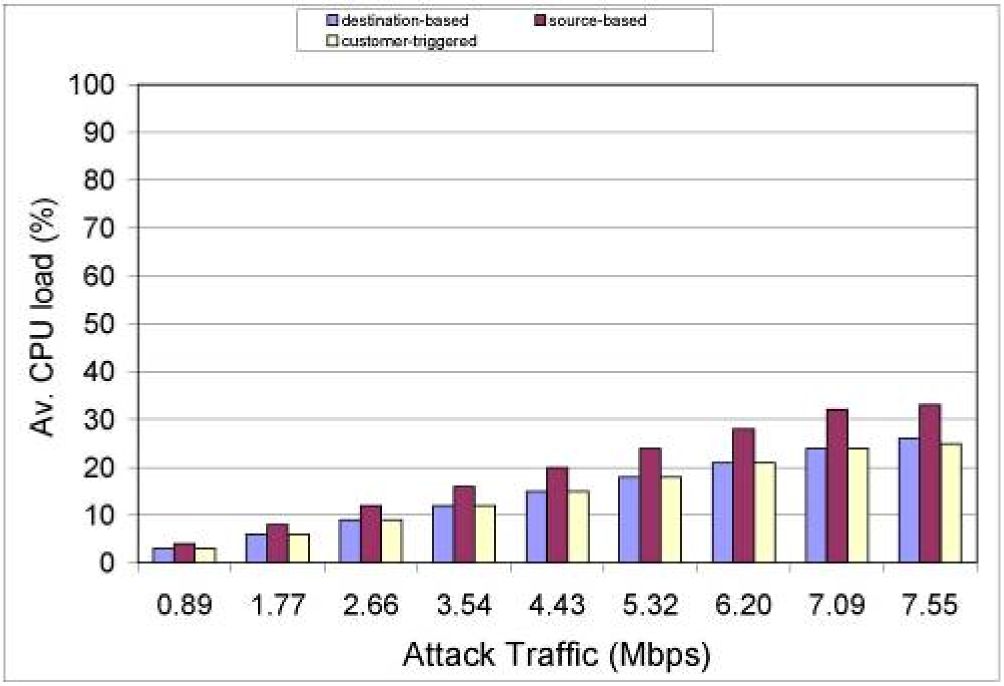


Figure 44. Border routers’ CPU load before application of BH routing

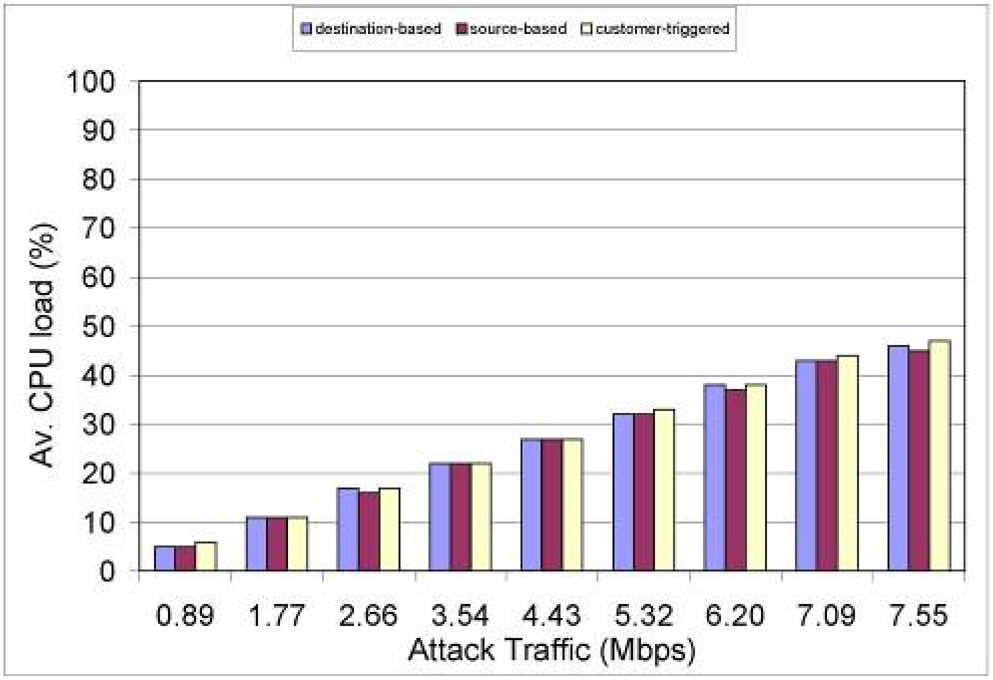


Figure 45. Router 1 CPU load before application of BH routing

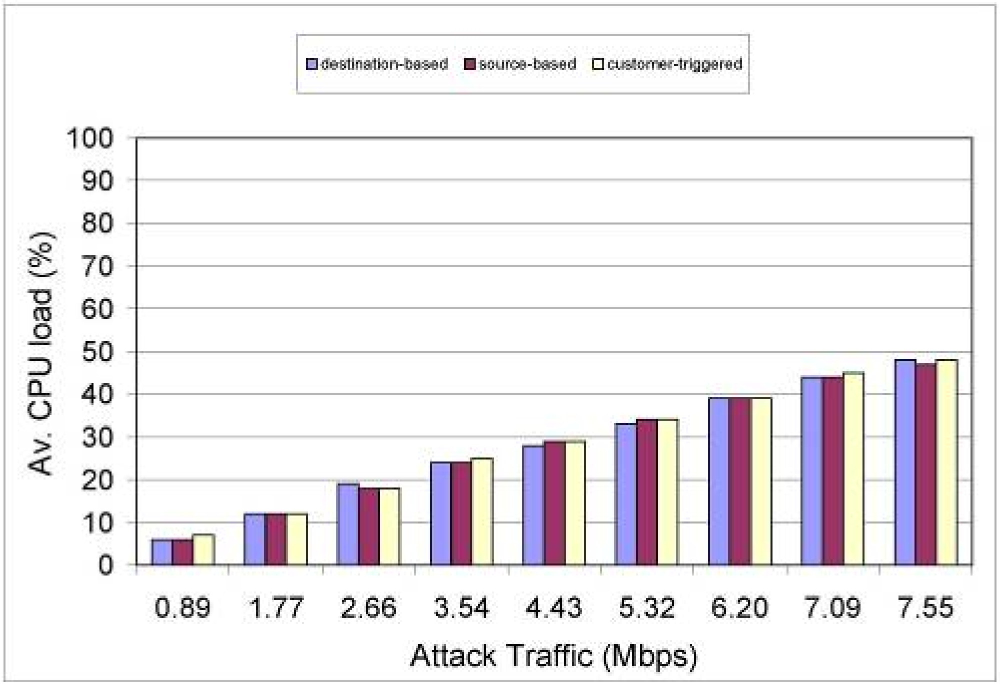


Figure 46. Router 2 CPU load before application of BH routing

Figures 45 and 46 show that the internal routers had the same CPU load in every Blackhole routing method. This result was expected since the configuration of those routers didn’t need to change for the Blackhole routing. Figure 44 also shows, however, that the border routers had an increased CPU load for the same traffic when they were configured to apply source-based RTBH routing. The percentage of the increased CPU load was 33.33 percent, which remained constant for all variations of the attack traffic.

The reason for the significant differences in the CPU load is that unicast Reverse Path Forwarding (uRPF) was applied in every border router when source-based Blackhole routing was used. Depending on the entries in the uRPF table, the percent of the increased CPU load may vary. We assumed that, before an application of Blackhole routing, most network administrators would keep the uRPF table minimal, as we did. This assumption led us to conclude that, on average, the configuration of source-based Blackhole routing will create a constant 33.33 percent larger CPU load on border routers. This is a significant factor that network administrators must consider before applying this method.

Since uRPF affects the CPU load before the application of Blackhole routing, intuition suggests that there should also be some kind of affect on the CPU load after the application of Blackhole routing. We therefore measured the border routers’ CPU load under different attack flows after the successful application of both source- and destination-based RTBH routing. Figure 47 shows the data collected from the Right border router in test-bed network #1.

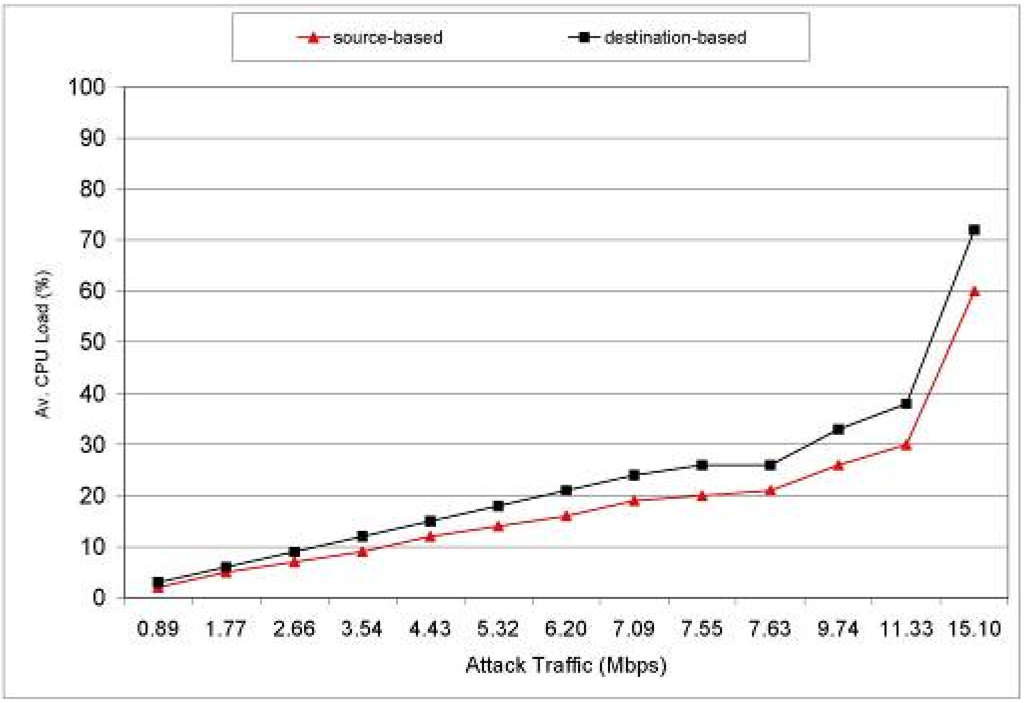


Figure 47. Right border router CPU load after application of source- and destination-based RTBH routing

Figure 48 shows the percentage of increase in the CPU load of the destination-based BH method as compared to the source-based BH method. The average increase was 28.5 percent.

The reason for this difference in CPU load is, again, the uRPF, but this time it works in favor of the source-based BH routing method. Since the uRPF process examines the packets first and the entries in the uRPF table are very few (just one in our simulation), the process of dropping packets with the source-based BH is faster than with the destination-based BH. Thus the CPU load is lower for the source-based BH. This is another significant factor that affects the choice of the most appropriate BH method to be used.

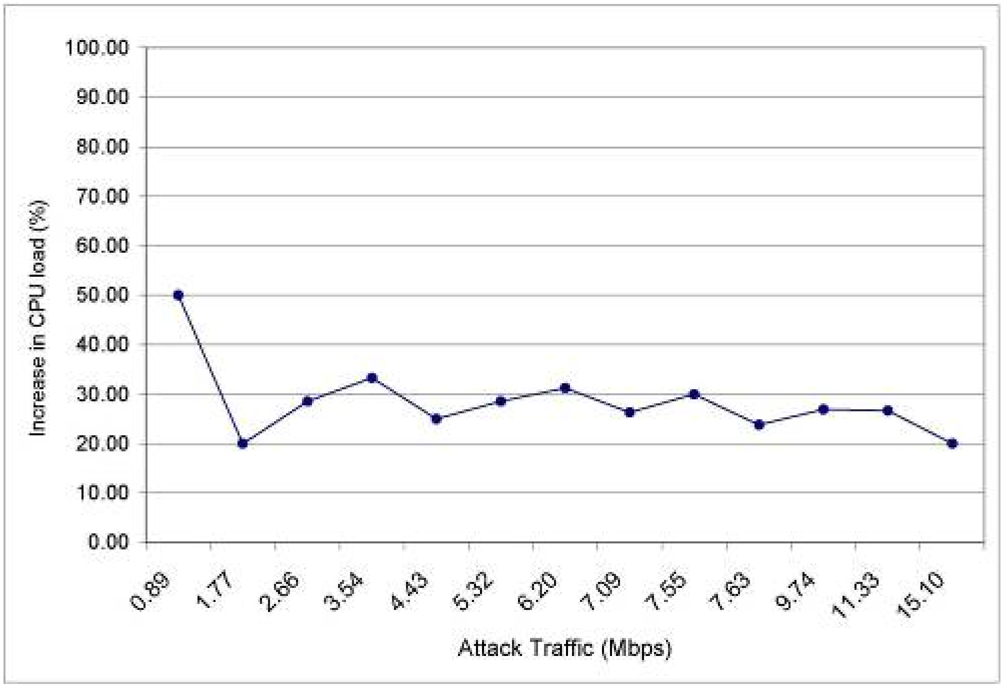


Figure 48. Percentage of increased CPU load in destination-based method as compared to the source-based method

In the next chapter we summarize the results presented in this chapter and propose potential future research topics related to this Master’s Thesis.

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONCLUSIONS AND FUTURE WORK

A. CONCLUSIONS

Using real test-bed networks, this study evaluated the performance of three BGP Blackhole routing methods in terms of their response time. The networks were put under stressful situations in which either the router CPU capacity or the link capacity for the iBGP session became a bottleneck. This evaluation produced the following conclusions:

First, we concluded that, of the two limiting factors inside a network, the CPU load and the link load, only the link load greatly degrades the performance of BGP Blackhole routing. We demonstrated, for the first time, that the router response time in high-link-load situations boosts to infinity. In practical terms, this means that BH routing is ineffective in defending against DDoS attacks in cases of high link loads. On the other hand, a high CPU load increased the response time from milliseconds to minutes, but in all our simulations the iBGP messages from the trigger were able to finally reach the border routers.

Second, we concluded that, of the three basic BGP Blackhole routing methods, the customer-triggered method has the worst performance. Our simulations for this method showed that, under low-to-medium attack traffic, the router response time reached values of more than 200 percent on average, compared to the average of the other two methods. Furthermore, for the same method under high attack traffic, the response time reached values of almost 100 percent compared to the destination-based BH routing.

Third, we found that, the best of the three methods is the destination-based BH routing. In almost all of the simulations, this method showed better performance in both response time and in the CPU load. The only exception to this occurred when we compared the CPU load after successful application of the BH routing. In that case, the source-based BH routing had a better CPU load by 28.5 percent on average.

The source-based BH routing’s standing falls between the other two methods, but is closest to the destination-based method. In low-to-medium attack traffic it showed the same performance in terms of the CPU load, but in high attack traffic the same metric had slightly higher values. A significant disadvantage of this method is that, by applying uRPF in a router’s interface, under normal operation the CPU load is increased.

The last conclusion of this research is that the timing of the application of the BGP Blackhole routing is very important in the case of either high link load or high CPU load. Especially in the high link load case the simulations showed that employment of BGP Blackhole would be totally inefficient if applied 40 seconds or more after the DDoS attack initialization.

B. FUTURE WORK

Our thesis results provide new opportunities and space for further study. Research in the following areas will provide more complete knowledge about BGP Blackhole routing.

1. As noted previously in this study, we assumed that a given DDoS attack had been positively identified by either an automated system or a human operator. The ability

to automatically identify an attack using an IDS/IPS system would greatly improve the performance of BGP Blackhole routing. As the research in this field to date is limited, it is our first suggested area for future work.

2. In this study we mainly used medium performance Cisco routers and only one Juniper router. But the probability that in a large network, routers of both vendors will co-exist is very high. Thus, another potential area of future study could involve evaluating the degree of interoperability of routers from different vendors that must all work together to apply BGP Blackhole routing.

3. Finally, this study showed that, due to the principles of TCP, OSPF, and BGP protocols in high link loads, the performance of BGP Blackhole routing is very degraded. More research in this area, therefore, would be very useful in showing how best to bypass the limitations of those protocols and in providing a solution that would enable BGP Blackhole routing to effectively mitigate DDoS attacks.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. CONFIGURATION FILES: TEST-BED NETWORK #1 ROUTERS

Appendix A presents the configuration files for test-bed network #1’s routers: one internal router, one border router, and the trigger router.

Internal Router #1:

```
!
! Last configuration change at 06:53:45 UTC Sun Mar 14 1993
! NVRAM config last updated at 06:54:26 UTC Sun Mar 14 1993
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router1
!
boot-start-marker
boot-end-marker
!
no logging console
!
no network-clock-participate slot 1
no network-clock-participate wic 0
no aaa new-model
ip subnet-zero
ip cef
!
!
interface Loopback0
 no ip address
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Ethernet1/0
 ip address 192.168.4.1 255.255.255.0
 full-duplex
!
interface Ethernet1/1
 ip address 192.168.6.1 255.255.255.0
 full-duplex
!
interface Ethernet1/2
 ip address 192.168.3.1 255.255.255.0
 full-duplex
!
interface Ethernet1/3
 ip address 192.168.7.1 255.255.255.0
 full-duplex
!
router ospf 200
 log-adjacency-changes
 redistribute connected
 network 192.168.0.0 0.0.255.255 area 0
!
no ip http server
ip classless
!
!
snmp-server community nikos RW
!
```

```
line con 0
 exec-timeout 35791 0
line aux 0
line vty 0 4
 login
!
ntp clock-period 17179859
ntp server 192.168.100.2
!
end
```

Left border router:

```
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname BR_Left
!
boot-start-marker
boot-end-marker
!
no logging console
!
no network-clock-participate slot 1
no network-clock-participate wic 0
no aaa new-model
ip subnet-zero
ip cef
!
!
interface Loopback0
 no ip address
!
interface Null0
```

```
 no ip unreachables
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Ethernet1/0
 no ip address
 shutdown
 full-duplex
!
interface Ethernet1/1
 ip address 192.168.2.1 255.255.255.0
 full-duplex
!
interface Ethernet1/2
 ip address 192.200.1.2 255.255.255.0
 ip verify unicast source reachable-via any
 full-duplex
!
interface Ethernet1/3
 no ip address
 shutdown
 half-duplex
!
router ospf 200
 log-adjacency-changes
 redistribute connected
 redistribute static
 redistribute bgp 100
 passive-interface Ethernet1/2
 network 192.168.0.0 0.0.255.255 area 0
!
router bgp 100
 no synchronization
 bgp log-neighbor-changes
 network 192.168.0.0 mask 255.255.0.0
 neighbor 192.168.100.2 remote-as 100
 no auto-summary
!
no ip http server
ip classless
ip route 192.0.2.0 255.255.255.0 Null0
!
!
logging trap debugging
logging facility local0
logging source-interface Loopback0
logging 192.168.5.2
snmp-server community nikos RW
!
line con 0
 exec-timeout 35791 0
line aux 0
line vty 0 4
 login
!
ntp clock-period 17180042
ntp server 192.168.100.2
!
end
```

Trigger router:

```
!
! Last configuration change at 07:57:00 UTC Thu Mar 4 1993
! NVRAM config last updated at 07:57:45 UTC Thu Mar 4 1993
```

```
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname trigger
!
boot-start-marker
boot-end-marker
!
no logging console
!
no network-clock-participate slot 1
no network-clock-participate wic 0
no aaa new-model
ip subnet-zero
ip cef
!
!
interface Loopback0
 no ip address
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Ethernet1/0
 no ip address
 shutdown
 full-duplex
!
interface Ethernet1/1
 ip address 192.168.100.2 255.255.255.0
 full-duplex
!
interface Ethernet1/2
 no ip address
 shutdown
 full-duplex
!
interface Ethernet1/3
 no ip address
 shutdown
 full-duplex
!
router ospf 200
 log-adjacency-changes
 network 192.168.0.0 0.0.255.255 area 0
!
router bgp 100
 no synchronization
 bgp log-neighbor-changes
 redistribute static route-map StaticToBGP
 neighbor 192.168.2.1 remote-as 100
 neighbor 192.168.6.2 remote-as 100
 neighbor 192.168.7.2 remote-as 100
 no auto-summary
!
ip http server
ip classless
!
!
logging trap debugging
logging facility local3
logging source-interface Loopback0
logging 192.168.5.2
route-map StaticToBGP permit 10
 match tag 20
 set ip next-hop 192.0.2.1
 set local-preference 50
 set origin igp
 set community no-export
!
route-map StaticToBGP permit 20
!
snmp-server community nikos RW
!
line con 0
 exec-timeout 35791 0
line aux 0
line vty 0 4
 login
!
ntp master 5
!
end
```

APPENDIX B. CONFIGURATION FILES: TEST-BED NETWORK #2 ROUTERS

Appendix B presents the configuration file of the trigger router for test-bed network #2:

```
!
! Last configuration change at 09:49:19 UTC Thu Mar 4 1993
! NVRAM config last updated at 09:44:32 UTC Thu Mar 4 1993
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname trigger
!
boot-start-marker
boot-end-marker
!
no logging console
!
no network-clock-participate slot 1
no network-clock-participate wic 0
no aaa new-model
ip subnet-zero
ip cef
!
!
interface Loopback0
 no ip address
!
interface FastEthernet0/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Ethernet1/0
 ip address 192.168.200.1 255.255.255.0
 full-duplex
!
interface Ethernet1/1
 ip address 192.168.100.2 255.255.255.0
 full-duplex
!
interface Ethernet1/2
 no ip address
 shutdown
 full-duplex
!
interface Ethernet1/3
 no ip address
 shutdown
 full-duplex
!
router ospf 200
 log-adjacency-changes
 network 192.168.0.0 0.0.255.255 area 0
!
router bgp 100
 no synchronization
 bgp log-neighbor-changes
 redistribute static route-map StaticToBGP
 neighbor 192.168.2.1 remote-as 100
 neighbor 192.168.6.2 remote-as 100
 neighbor 192.168.7.2 remote-as 100
 no auto-summary
!
ip http server
ip classless
!
!
logging trap debugging
logging facility local3
logging source-interface Loopback0
logging 192.168.5.2
route-map StaticToBGP permit 10
 match tag 20
 set ip next-hop 192.0.2.1
 set local-preference 50
 set origin igp
 set community no-export
!
route-map StaticToBGP permit 20
!
snmp-server community nikos RW
!
line con 0
 exec-timeout 35791 0
line aux 0
line vty 0 4
 login
!
ntp master 5
!
end
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. CONFIGURATION FILES: TEST-BED NETWORK #3 ROUTERS

Appendix C presents the configuration file of the Juniper J4300 router for test-bed network #3:

```
      version 7.1R1.3;
      system {
          host-name j43;
          root-authentication {
              encrypted-password   "$1$EEuNurtm$oZk6BaPntQac9CNwiYUeK.";
## SECRET-DATA
          }
          login {
              user j43 {
                  uid 2000;
                  class superuser;
                  authentication {
                      encrypted-password
"$1$M0NSaWol$urTrVGSc7gBFZ64RNguyA/"; ## SECRET-DATA
                  }
              }
          }
          services {
              ssh;
              telnet;
              web-management {
                  http;
              }
          }
          syslog {
              user * {
                  any emergency;
              }
              file messages {
                  any any;
```

```
            authorization info;
        }
        file interactive-commands {
            interactive-commands any;
        }
    }
    ntp {
        server 192.168.100.2;
    }
}
interfaces {
    fe-0/0/0 {
        speed 10m;
        link-mode full-duplex;
        unit 0 {
            family inet {
                address 192.168.2.2/24;
            }
        }
    }
    fe-0/0/1 {
        speed 100m;
        link-mode full-duplex;
        unit 0 {
            family inet {
                address 192.168.200.1/24;
            }
        }
    }
    fe-1/0/0 {
        speed 10m;
        link-mode full-duplex;
        unit 0 {
            family inet {
                address 192.168.100.1/24;
            }
        }
```

```
}
##
## Warning: requires an additional 'if-fe' license
##
fe-1/0/1 {
    unit 0 {
        family inet {
            address 192.168.5.1/24;
        }
    }
}
fe-5/0/0 {
    speed 10m;
    link-mode full-duplex;
    unit 0 {
        family inet {
            address 192.168.7.2/24;
        }
    }
}
##
## Warning: requires an additional 'if-fe' license
##
fe-5/0/1 {
    speed 10m;
    link-mode full-duplex;
    unit 0 {
        family inet {
            address 192.168.6.2/24;
        }
    }
}
lo0 {
    unit 0 {
        family inet {
            address 127.0.0.1/32;
        }
        }
    }
}
snmp {
    community nikos {
        authorization read-write;
    }
}
protocols {
    ospf {
        area 0.0.0.0 {
            interface all;
        }
    }
}
```

LIST OF REFERENCES

Battles, Tim, McPherson, Danny and Morrow, Chris. "Customer-Triggered Real-Time Blackholes." NANOG. http://www.nanog.org/mtg-0402/pdf/morrow.pdf (accessed 09/14/2006).

Cisco. "REMOTELY TRIGGERED BLACK HOLE FILTERING—DESTINATION BASED AND SOURCE BASED." Cisco Press. http://www.cisco.com/warp/public/732/Tech/security/docs/blackhole.pdf (accessed 09/14/2006).

Claiborne, Anna. "Information Collection on DDoS Attacks." NANOG. http://www.nanog.org/mtg-0606/pdf/anna-claiborne.pdf (accessed 09/13/2006).

Gibson, Steve. "The Strange Tale of the Denial of Service Attacks on Grc.Com." GRC.com. http://www.grc.com/dos/grcdos.htm (accessed 09/14/2006).

Kleffman, Michael D. "Analysis of Effects of BGP Black Hole Routing on a Network Like the NIPRNET." Master's of EE, AFIT, 2005 (accessed 09/15/2005).

Mirkovic, Jelena, Dietrich, Sven, Dittrich, David and Reiher, Peter. "Understanding Denial of Service." Prentice Hall PTR. http://www.phptr.com/articles/article.asp?p=386163&seqNum=5&rl=1 (accessed 09/13/2006).

Raveendran Greene, Barry. "Remote Triggering Black Hole Filtering." Cisco Press. http://www.ispbook.com/supplements/Remote_Triggered_Black_Hole_Filtering-02.pdf (accessed 09/14/2006).

Security Scape. "Autonomic Systems-Combating DDoS Attacks." Security Scape. http://www.securesynergy.com/library/articles/037-2003.php (accessed 09/14/2006).

Vayner, Arie. "DoS Attacks on the Internet." Israeli Open University. http://www.vayner.net/dos/dos.html (accessed 09/14/2006).

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, VA

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, CA

3. Geoffrey Xie, xie@nps.edu
   Naval Postgraduate School
   Monterey, CA

4. J.D. Fulp, jdfulp@nps.edu
   Naval Postgraduate School
   Monterey, CA

5. Nikolaos Stamatelatos, nstamate@nps.edu
   Naval Postgraduate School
   Monterey, CA

6. Neal Ziring, nziring@thematrix.ncsc.mil
   National Security Agency
   Fort Meade, MD

7. Matthew N. Smith, m.smith@dewnet.ncsc.mil
   National Security Agency
   Fort Meade, MD

8. D. C. Boger, dboger@nps.edu
   Naval Postgraduate School
   Monterey, CA